

SKRIPTA RAZVOJ RAČUNALNIH IGARA

Autor: Nenad Breslauer, v. pred.

Čakovec, 2022.

Autor:

Nenad Breslauer, v. pred.

Recenzenti:

dr.sc. Oliver Jukić, prof.v.š.,

Doc. dr. sc. Zvonimir Sabati

Nakladnik: Međimursko veleučilište u Čakovcu

Za nakladnika: doc.dr.sc. Igor Klopotan, v. pred.

ISBN 978-953-8095-21-4

Copyright © Međimursko veleučilište u Čakovcu

PREDGOVOR

Skripta služi stjecanju osnovnih znanja unutar područja predmeta Razvoja računalnih igara.

Ova skripta za laboratorijske vježbe sastavni je dio gradiva izbornog predmeta Razvoj računalnih igara. Kroz poglavlja obrađuje gradivo o osnovama razvoja računalnih igara u Unity 3D okruženju. Pomoću konkretnih slika napravljenih u samim programima i popratnim tekstom trebala bi olakšati usvajanje gradiva.

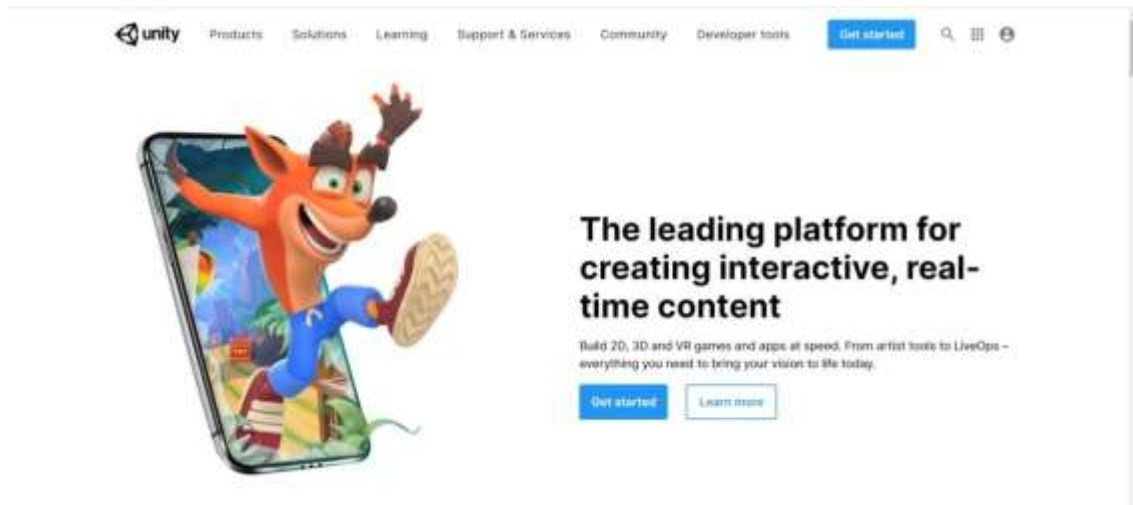
Gradivo je po sadržaju i redoslijedu izvođenja usklađeno sa sadržajem i redoslijedom izvođenja predavanja i auditornih vježbi iz navedenog predmeta.

U ovoj skripti ćete naučiti kako napraviti jednostavnu video igru. Prvi dio ove skripte uvod je u Unity za one koji ga nikad prije nisu koristili. Unity je besplatni program za izgradnju video igara i virtualne stvarnost (VR). Naučit ćete kako postaviti scenu, dodati predmete i promijeniti izgled vaše igre u Unityu.

U drugom dijelu ćemo iz temelja izgraditi potpuno funkcionalnu igru.

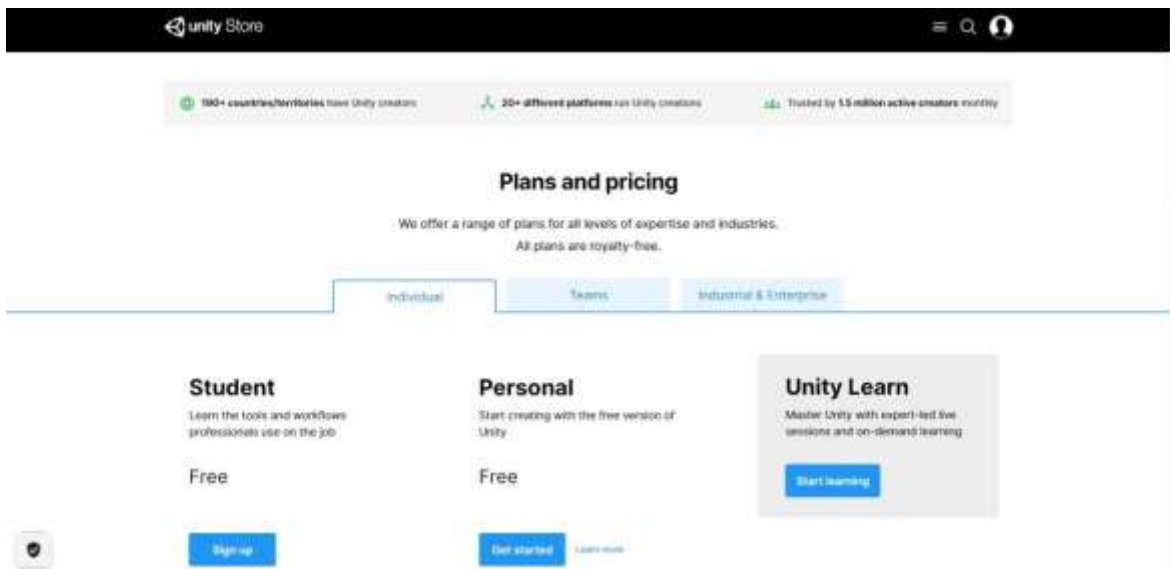
1. INSTALACIJA UNITY-a

Da biste preuzeli Unity, posjetite www.unity.com. Na početnoj stranici pritisnite gumb "Get Started".



Slika 1 Početna stranica Unity.com

Bit ćete preusmjereni na stranicu koja sadrži verzije Unity-a koje možete preuzeti. Osobna (eng.Personal) verzija je besplatna. S ovom verzijom možete objavljivati igre u App Storeu. Za daljnji tijek instalacije Kliknite gumb "Get Started" pod "Personal" i slijedite korake.



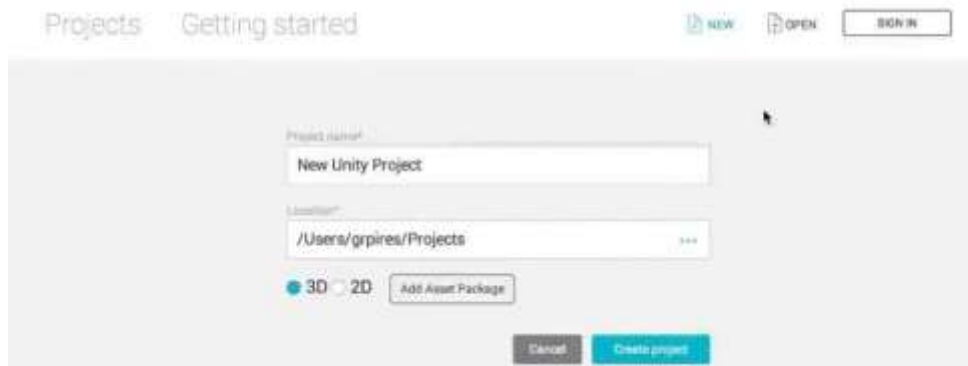
Slika 2 Unity store

2. UVOD U UNITY

Stvaranje novog projekta

Da biste napravili novi projekt u Unityju, kliknite “New” u gornjem desnom dijelu početnog zaslona Unity-a.

Na početnom zaslonu možete prilagoditi neke postavke projekta.



Slika 3 Početni zaslon

Dajte projektu naziv i mjesto. Polje Location odnosi se na mapu u koju želite spremiti projekt. Dobra je praksa stvoriti novu mapu samo za svoje Unity projekte kako biste ih kasnije lakše pronašli.

Omogućite način 3D prikazivanja. Pritisnite gumb "Create project". Unity će stvoriti mapu s istim imenom kao i naziv vašeg projekta na mjestu koje ste naveli.

Otvorit će se Unity Editor. Zaslon će izgledati ovako:



Slika 4 Izgled Unity Editora

Lijeva bočna traka je prozor Hijerarhije. To je poput hijerarhije koju imate u svom operativnom sustavu. Na primjer, u mapi Slike možete imati podmape događaja koje grupiraju fotografije.

Hijerarhija i prizor

Hijerarhija navodi imena predmeta na sceni igre. Svaki objekt može imati svoju hijerarhiju. Trenutno imamo stavke Glavna kamera (*eng. Main Camera*) i Usmjereno svjetlo (*eng. Directional light*), koji su **objekti igre**. Sve što promijeni način na koji igra djeluje vizualno ili iznutra, poput igrača, neprijatelja ili eksplozije, treba biti u objektu igre.

Objekte navedene u Hijerarhiji možete vidjeti u prozoru Scene, a to je prozor u središtu vašeg zaslona. Scena prikazuje virtualni svijet u kojem gradimo igru.

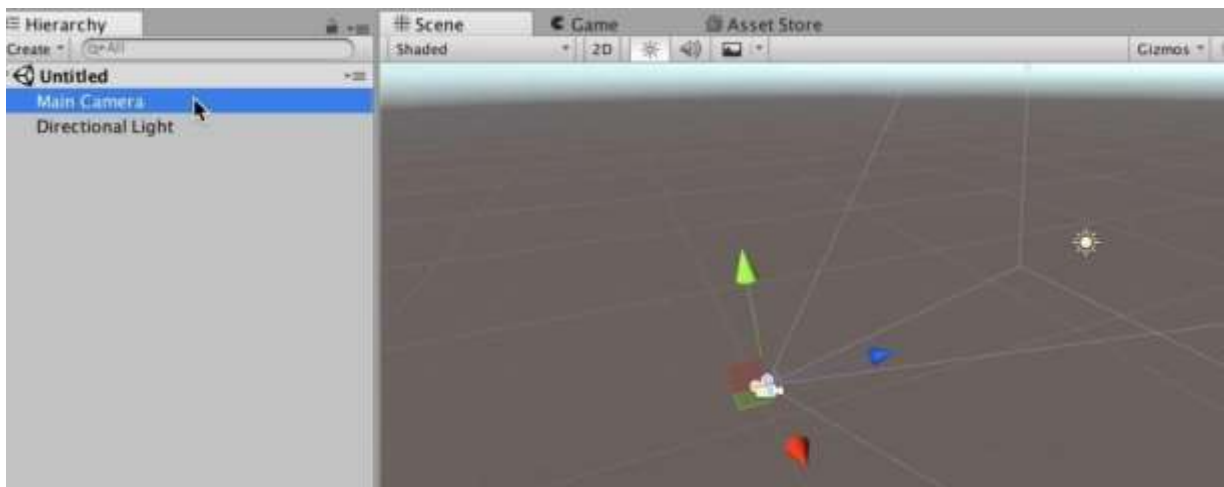
Pritiskom desne tipke miša u prozoru kursor će se promijeniti u ikonu oka, što je vidljivo na sljedećoj slici. Držeći desnu tipku miša, možete pomicati kursor da biste pogledali Scenu.



Slika 5 Izgled Scene

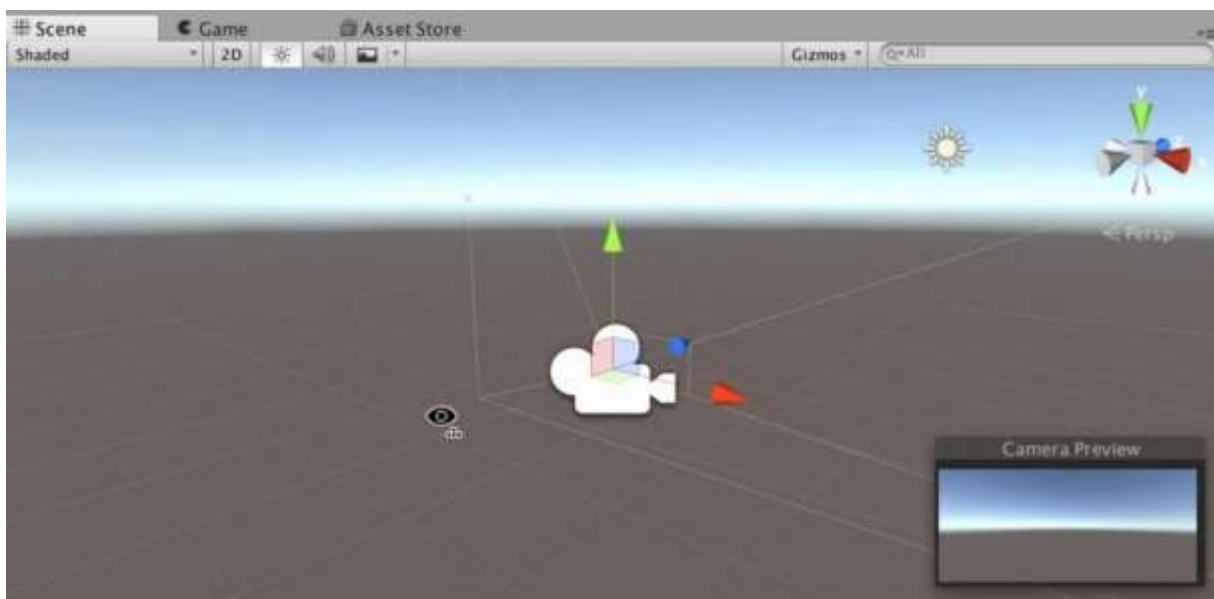
Drugi način kretanja scenom je pritiskanje tipki W, A, S i D na tipkovnici držeći desnu tipku miša.

Odabirom "Main Camere" u hijerarhiji. Strelice će se pojaviti oko simbola kamere u Scene.



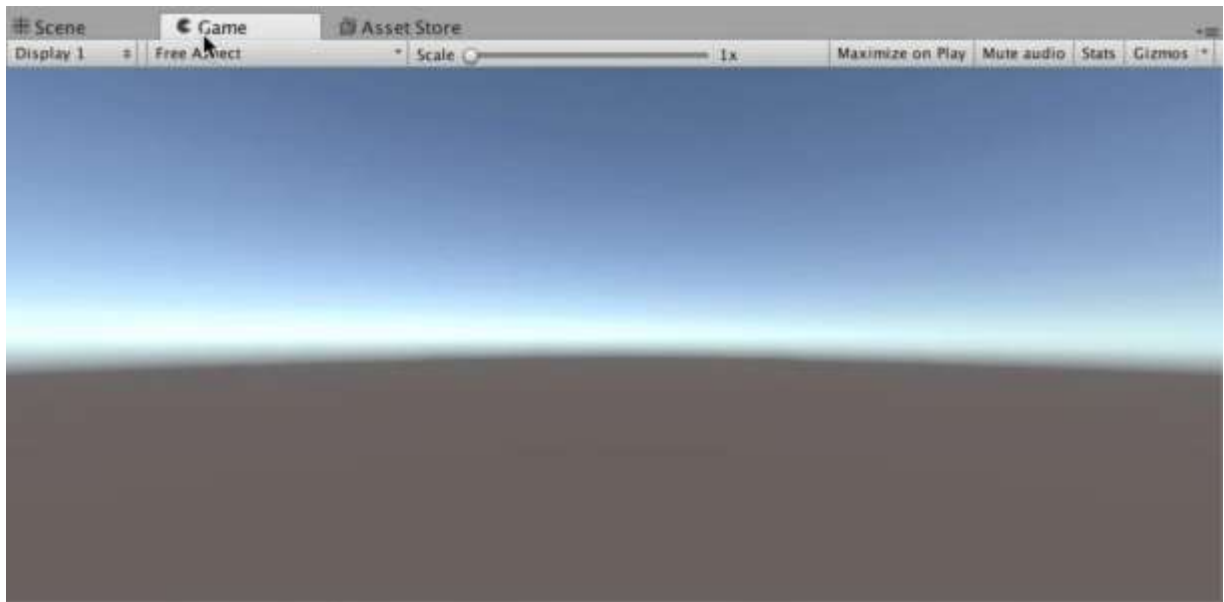
Slika 6 Izgled Main Camere

Glavna kamera prikazuje (crta) igru za igrača. Dok je odabrana glavna kamera, u donjem desnom dijelu scene nalazi se okvir za pregled kamere (eng. *Camera Preview*). Pregled kamere prikazuje scenu iz perspektive igrača.



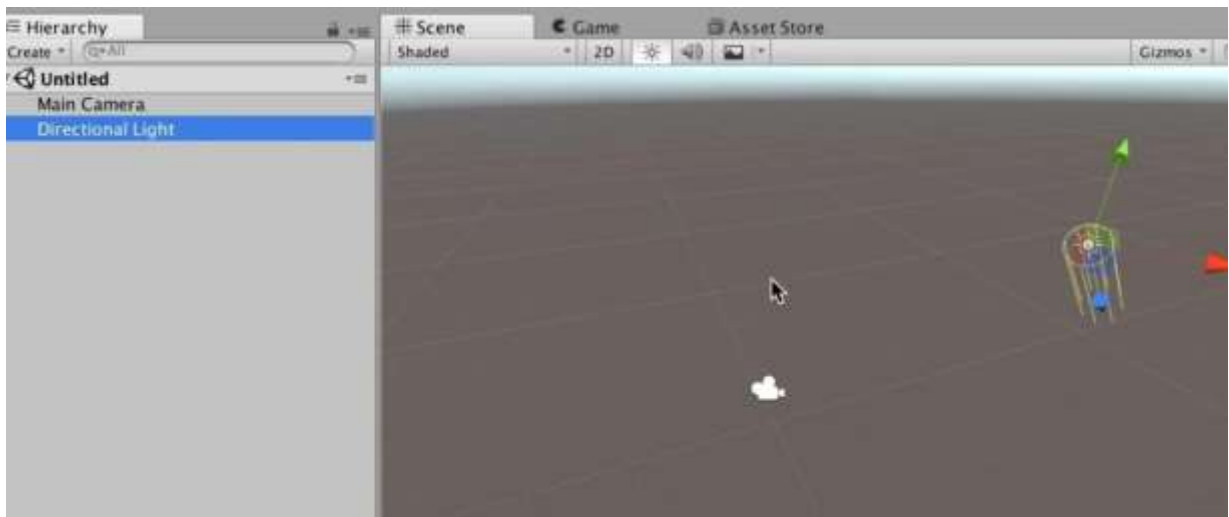
Slika 7 Camera Preview

Da biste vidjeli scenu s gledišta igrača, kliknite karticu "Game" pored kartice "Scena". Otvorit će se prozor Igre, kao na sljedećoj slici:



Slika 8 Izgled Game scene

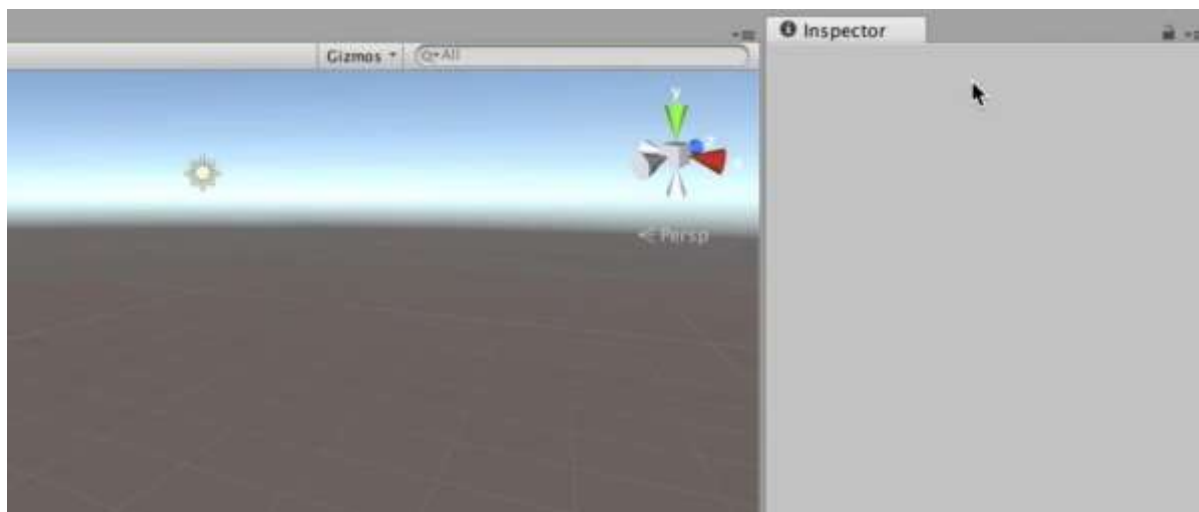
Ponovo otvorite prozor Scene. Kliknite "Directional Light" u hijerarhiji. Strelice će se pojaviti oko simbola usmjerenog svjetla (*eng. Directional Light*) u sceni. **Usmjereno svjetlo emitira svjetlosne zrake.** Ovaj objekt igre možete promijeniti da biste promijenili osvjetljenje scene.



Slika 9 Izgled Direction Light

Inspektor

Prozor s desne strane Scene sadrži inspektora. Ovaj prozor prikazuje detalje o odabranoj stavci, kao što je objekt igre u Sceni ili datoteka u našem projektu.



Slika 10 Inspector prozor

Kliknite na glavnu kameru. Inspektor će ispuniti atribute glavne kamere koje možete promijeniti. Na vrhu je naziv predmeta. Kartica Tag omogućuje vam označavanje objekta.



Slika 11 Inspector glavne kamere

Inspektor glavne kamere prikazuje sljedeće komponente: Transform, Camera, GUI Layer, Flare Layer i Audio Listener. Komponente mijenjaju ponašanje objekata igre.

Komponenta Transform najosnovnija je komponenta objekta igre. Transform vam omogućuje postavljanje položaja, rotacije i razmjera objekta.



Slika 12 Transform glavne kamere

Komponenta kamere prikazuje detalje o kameri. Trenutno je svojstvo Clear Flags "Skybox", tako da Game Window sadrži nebo. Ako promijenite Clear Flags u "Solid Color", prozor Game ispunit će se bojom svojstva "Background".



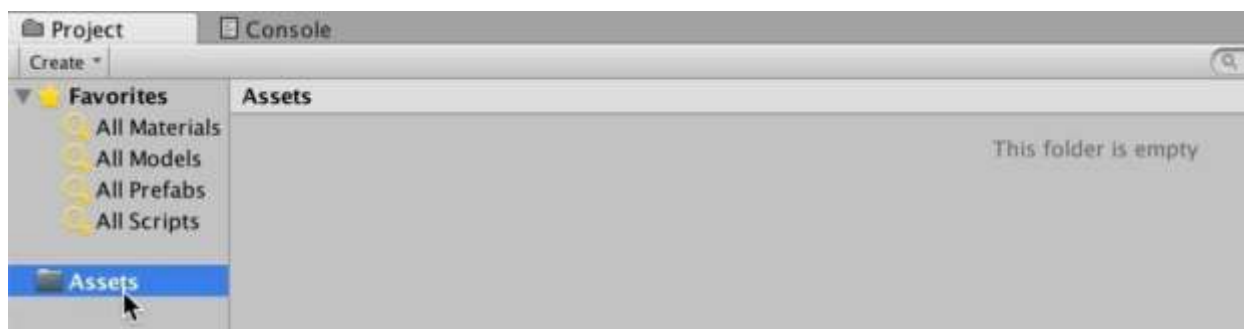
Slika 13 Postavke kamere

Odaberite usmjereno svjetlo iz hijerarhije. Inspektor će se promijeniti tako da sadrži komponente predmeta Usmjereno svjetlo. U komponenti Svjetlost možete promijeniti vrstu svjetla, boju, intenzitet, sjene i još mnogo toga.



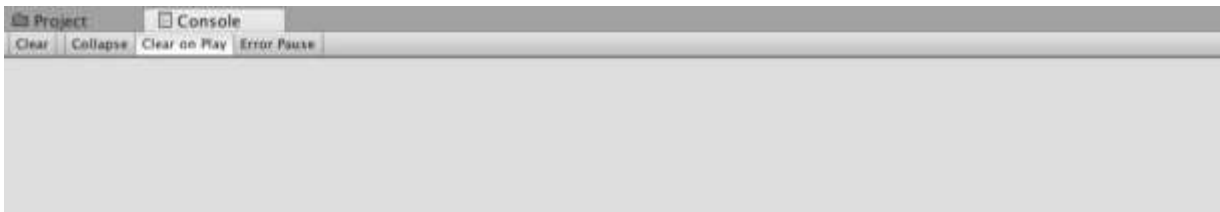
Slika 14 Postavke svijetla

Donji prozor u Unity sadrži prozor Project. Ovaj prozor sadrži mapu Assets koja sadrži sve datoteke koje čine vašu igru. Bilo koji 3D modeli, skripte, materijali, animacije, zvučni isječci, video datoteke bit će u Assets- u (imovini).



Slika 15 Assets prozor

Pored kartice Projekt nalazi se kartica Konzola (eng. Console). Imajte na umu da ako "Konzola" nije omogućena uz "Projekt" na vašem zaslonu, idite na Unity> Window> Console.



Slika 16 Prozor Console

Konzola prikazuje poruke vama kao programeru. Mogu se pojaviti tri vrste poruka:

- **Poruke o pogreškama** - obojene crveno; ikona je osmerokut. Poruke o pogreškama mogu se pojaviti kada u vašem kodu postoji pogreška ili kada dodatak ne radi.
- **Poruke upozorenja** - obojene crveno; ikona je trokut. Upozorenja mogu pružiti prijedloge koji poboljšavaju vaš kôd.
- **Osnovne informativne poruke** - bijele ili crne boje; ikona je govorni balon. U kodu igre možete postaviti poruke za ispis na konzoli. Te vas poruke mogu obavijestiti o tome što se događa u igri kako biste znali ponaša li se igra ispravno.

2. Manipulacija objektima

Pozicija

Odaberite glavnu kameru iz Hierarhije. Da biste promijenili X položaj kamere u Sceni, možete promijeniti "0" u polju X svojstva Position. Na primjer, promijenite X položaj glavne kamere na 2, kao na sljedećoj slici. Kamera će se pomaknuti u Sceni.



Slika 17 Postavljanje pozicije kamere

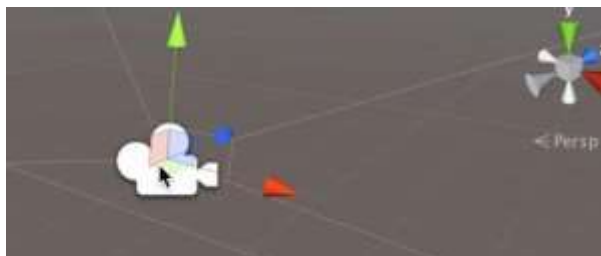
Drugi način promjene položaja X je klikanje na "X" pored "Položaj" i povlačenje kursora lijevo i desno.

Primijetite da promjena položaja X pomiče kameru u jednom stupnju slobode: duž vodoravne osi. Horizontalna os prikazana je kao crvena strelica koja se proteže od kamere u Sceni. Možete kliknuti na ovu crvenu strelicu za pomicanje kamere vodoravno.

Gizmo u gornjem desnom dijelu prozora Scene pokazuje da se crvena strelica odnosi na X os.

Položaj Y predstavlja vertikalnu os. Promjena položaja Y u komponenti Transform pomiče kameru gore-dolje.

Položaji X i Y omogućuju nam da kameru postavimo u 2D okruženje. Položaj Z, koji predstavlja os dubine, stvara 3D okruženje. Promjenom položaja Z pomičete kameru naprijed i natrag.



Slika 18 Postavljanje pozicije kamere

Uvećajte glavnu kameru u sceni. Tri pravokutnika nalaze se na vrhu ikone kamere. Možete kliknuti i povući pravokutnike za pomicanje kamere u različitim ravninama. Na primjer, zeleni pravokutnik pomiče kameru u ravnini XZ.

Rotacija

Svojstvo Rotation u komponenti Transform djeluje slično položaju. Promjenom X rotacije kamera gleda od gore ili dolje. Promjenom Y rotacije kamera gleda lijevo i desno. Promjena položaja Z okreće kameru ustranu, kao da igrač naginje glavu ulijevo i udesno.

Postoji još jedan način za promjenu rotacije. U gornjem lijevom kutu prozora Unity nalazi se nekoliko ikona koje se odnose na dostupne alate. Prvi alat pomiče razvojnu kameru. Ovaj je alat alternativa držanju desne tipke miša.

Drugi alat, koji je unaprijed odabran prema zadanim postavkama, omogućuje vam premještanje objekata. Klikom na treću ikonu mijenjaju se alati oko glavne kamere, što je vidljivo na sljedećoj slici. Umjesto strelica bit će crveni, zeleni i plavi krug.

Ako kliknete i povučete krugove, kamera će se zakrenuti. Boje odgovaraju različitim osima: crvena za X, zelena za Y, plava za Z. Ako kliknete na kuglu između krugova, možete zakrenuti kameru u sve tri osi.



Slika 19 Rotiranje kamere

Skaliranje

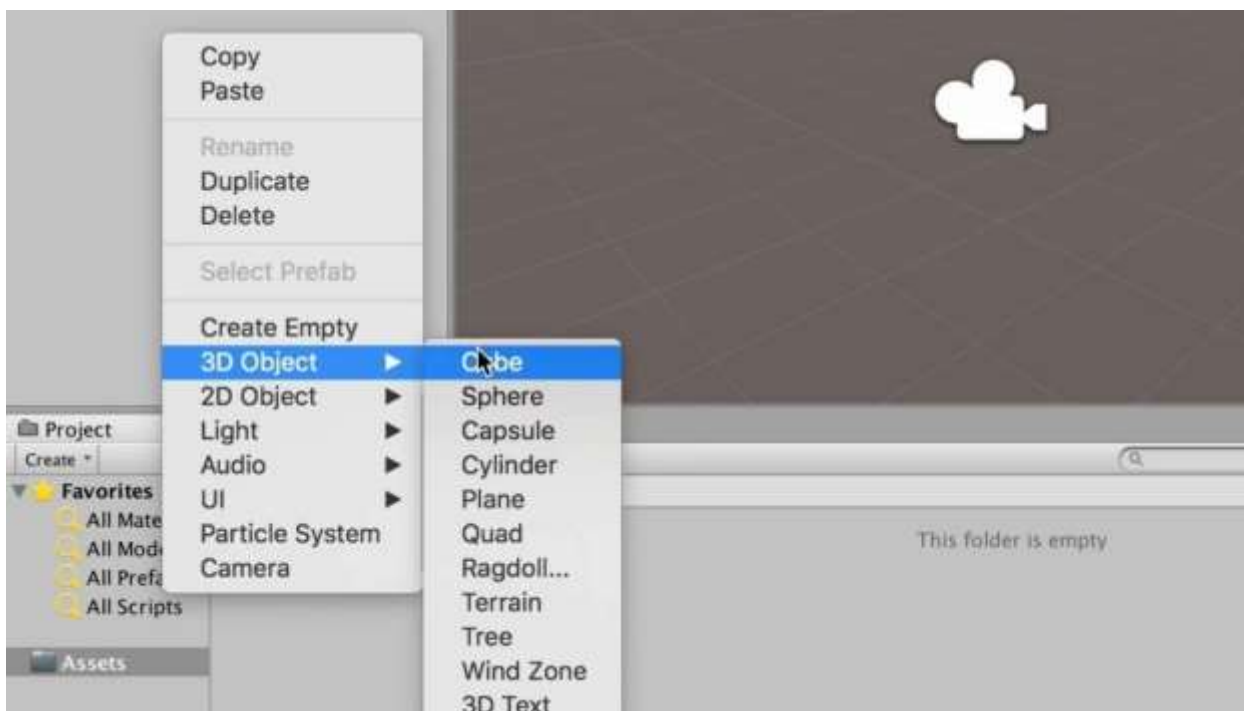
Treći element u komponenti Transform je Scale, koji također ima vrijednosti X, Y i Z. Razmjerno mijenja veličinu objekta. **Ako promijenite vrijednosti skale za glavnu kameru, veličina kamere neće se promijeniti.** To je zato što se ikona kamere ne prikazuje u igri. Ikona predstavlja gledište igrača.

Usmjereno svjetlo je također apstraktno. Uređaj neće vidjeti svoju ikonu. Tako Scale neće raditi na tim objektima. Međutim, Scale će promijeniti veličinu 3D elementa.

Da biste prototipirali igru u Unityju, možete dizajnirati scenu pomoću primitiva. Postoji nekoliko primitiva ili elemenata koje možete stvoriti u Unityju, poput kocke.

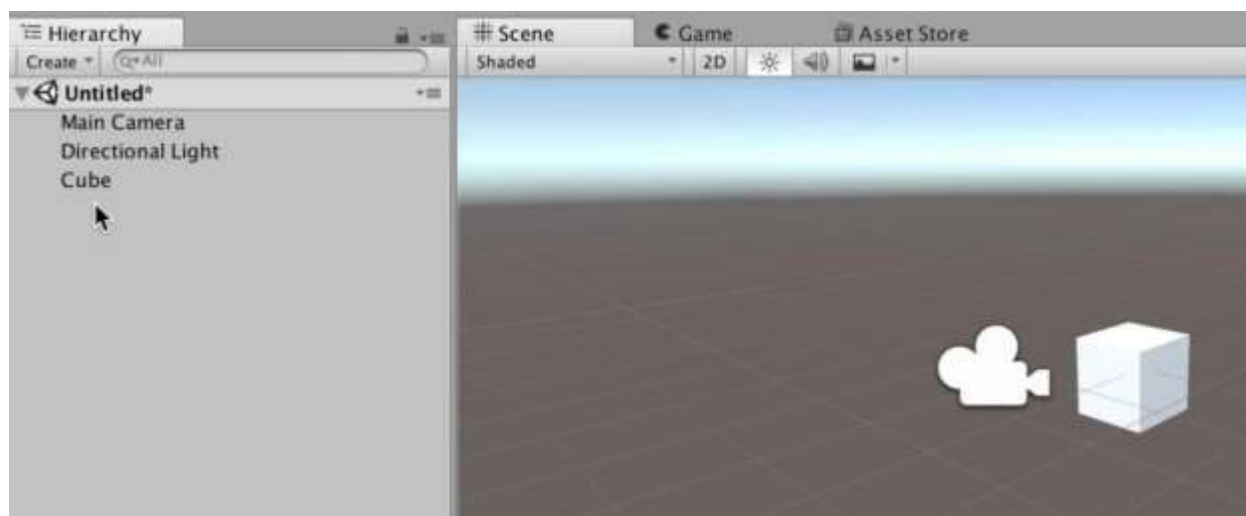
Da biste napravili 3D element, kliknite desnu tipku miša u Hijerarhiji. Pojavit će se izbornik koji sadrži popis objekata igre koje možete stvoriti. Klikom na "Create Empty" stvorit će se mrežni objekt koji ima samo komponentu Transform.

Na kartici "3D objekt" nalaze se nazivi trodimenzionalnih oblika. Kliknite "Cube", kao na sljedećoj slici.



Slika 20 Stvaranje kocke

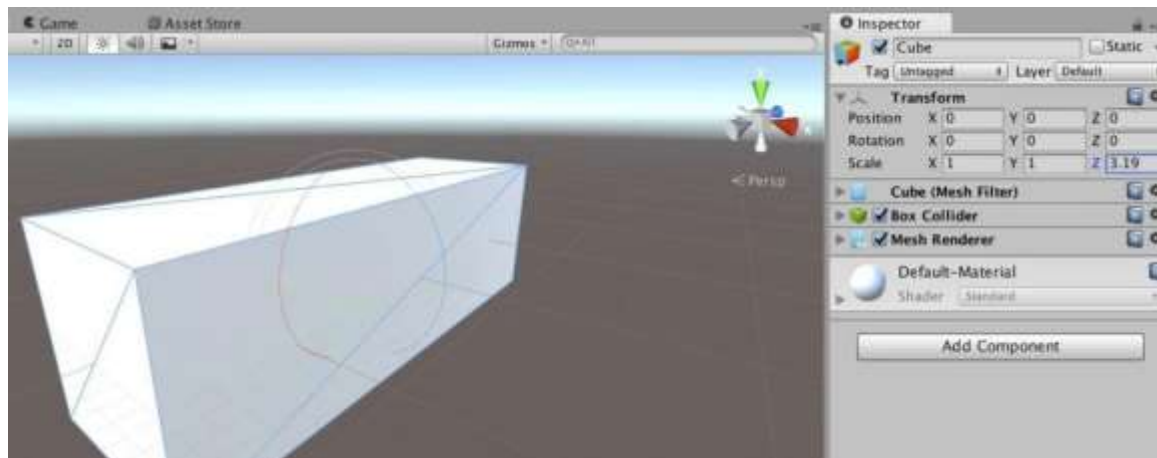
"Kocka" će se pojaviti u Hijerarhiji. Kocka će se pojaviti u Sceni, kao što možete vidjeti na sljedećoj slici.



Slika 21 Kocka u Sceni

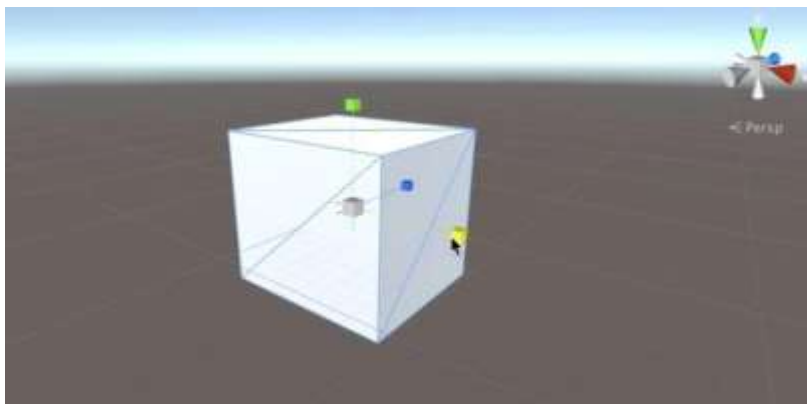
U Inspektoru postoje mnoge komponente za Kocku. Da biste postavili kocku u središte scene, promijenite vrijednosti X, Y i Z Position na 0.

Ako promijenite vrijednost kocke X Scale, veličina kocke će se promijeniti na osi X. Promjenom Y vrijednosti skale kocka postaje viša ili kraća. Vrijednost Z ljestvice odgovara veličini kocke u osi dubine, što je vidljivo na sljedećoj slici.



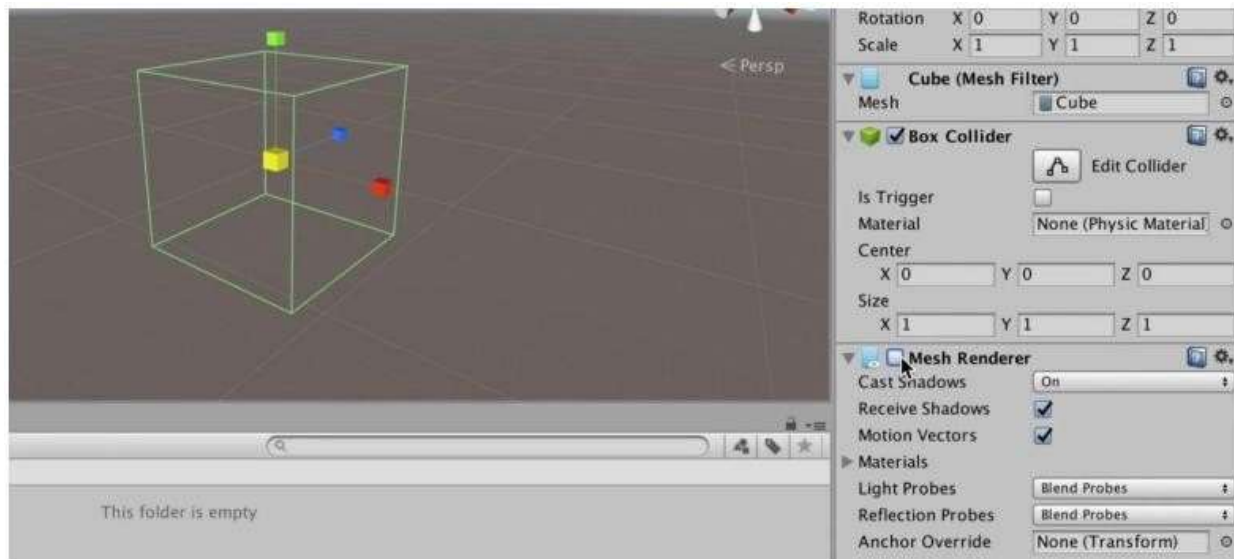
Slika 22 Skaliranje kocke

Da biste vizualno promijenili skaliranje Kocke, kliknite četvrtu ikonu u gornjem lijevom kutu Unityja. Kocke u boji pružaju se od središta Kocke. Možete povući kocke da biste promijenili veličinu Kocke. Ako kliknete i povučete sivu kocku u središte objekta kocke, veličina Kocke promijenit će se u sve tri osi odjednom.



Slika 23 skaliranje kocke

Komponenta kocka (mrežasti filter) podaci su za 3D objekt koji se prikazuje na sceni. Box Collider koristi se za obradu sudara.



Slika 24 Box Collider kocke

Mesh Renderer crta kocku. Ako onemogućite Mesh Renderer uklanjanjem oznake, kocka će postati nevidljiva.

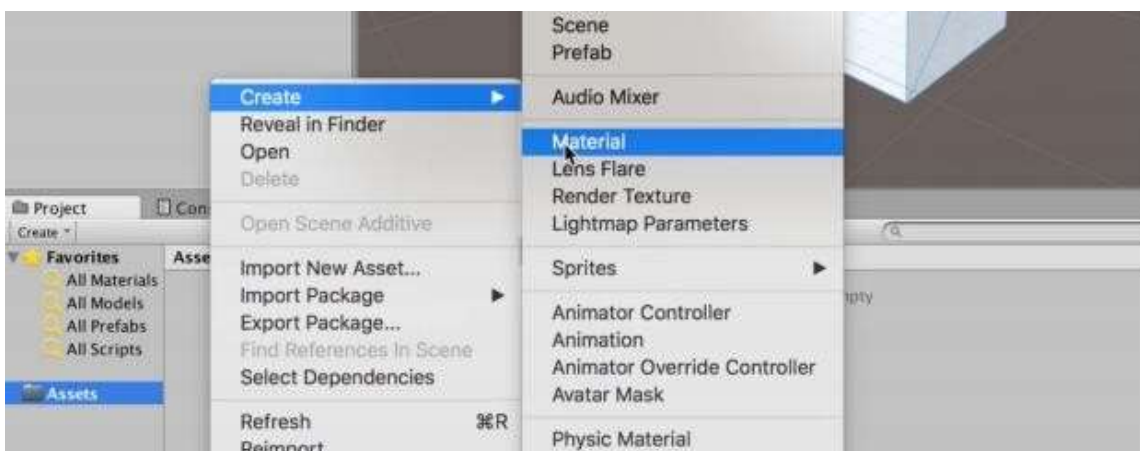
4. MATERIJALI

Mesh Renderer crta objekte igre u Sceni. Ova komponenta sadrži nekoliko svojstava. Na primjer, Cast Shadows definira hoće li se sjene crtati iz kocke. Receive Shadows određuje hoće li se u tome crtati sjene ostalih elemenata kocke.

4.1. Mijenjanje Boja

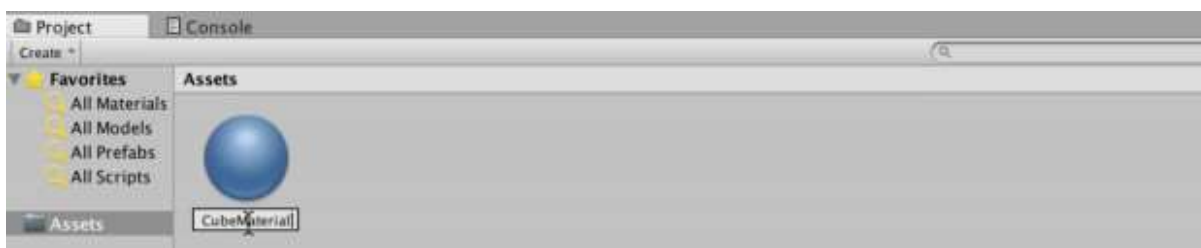
Što je materijal? Pretpostavimo da želimo promijeniti boju Kocke. Unity pruža zadani materijal "Default-Material" za kocku. Da bismo promijenili boju kocke, moramo stvoriti vlastiti materijal.

Desnom tipkom miša kliknite u mapu Assets u prozoru Project. Odaberite Create> Material, kao što je prikazano na slici ispod.



Slika 25 Izrada Materijala

Kao što je vidljivo na sljedećoj slici, materijal će se pojaviti u Assets-u. Postavite naziv materijala na "CubeMaterial".



Slika 26 Materijal kocke

Inspektor će sadržavati atribut CubeMaterial. Kliknite bijeli pravokutnik u odjeljak "Main Maps" pod "CubeMaterial". Otvoriti će se alat za odabir boja, što možete vidjeti na sljedećoj slici.

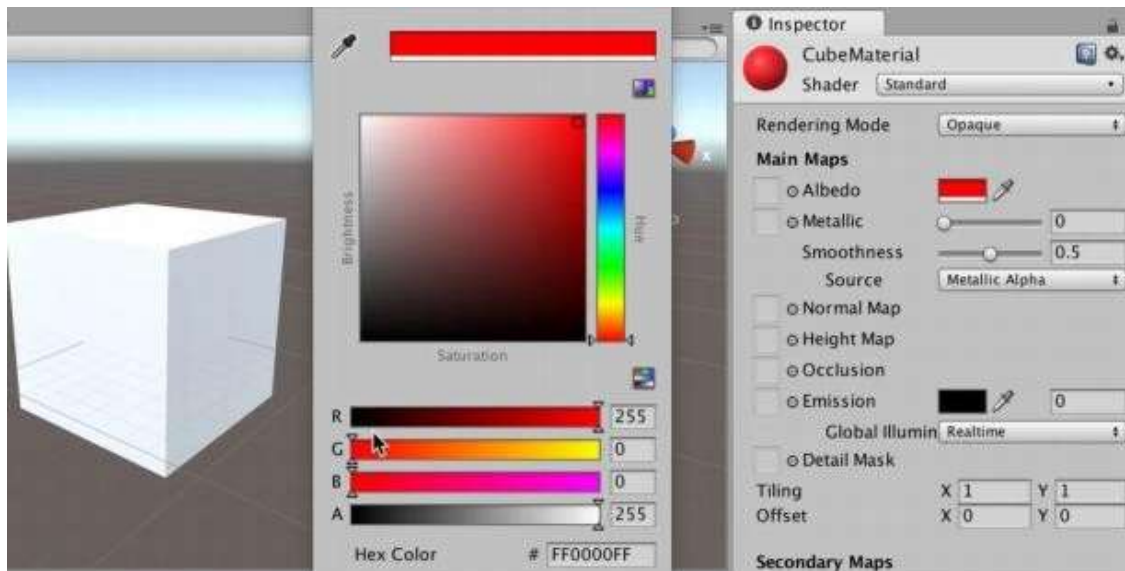


Slika 27 Main Maps

Odaberite boju, poput crvene. Vrijednosti klizača s oznakama "R", "G", "B" i "A" će se promijeniti. To je zato što se svaka boja može predstaviti kao kombinacija četiri vrijednosti: crvena, zelena, plava i alfa.

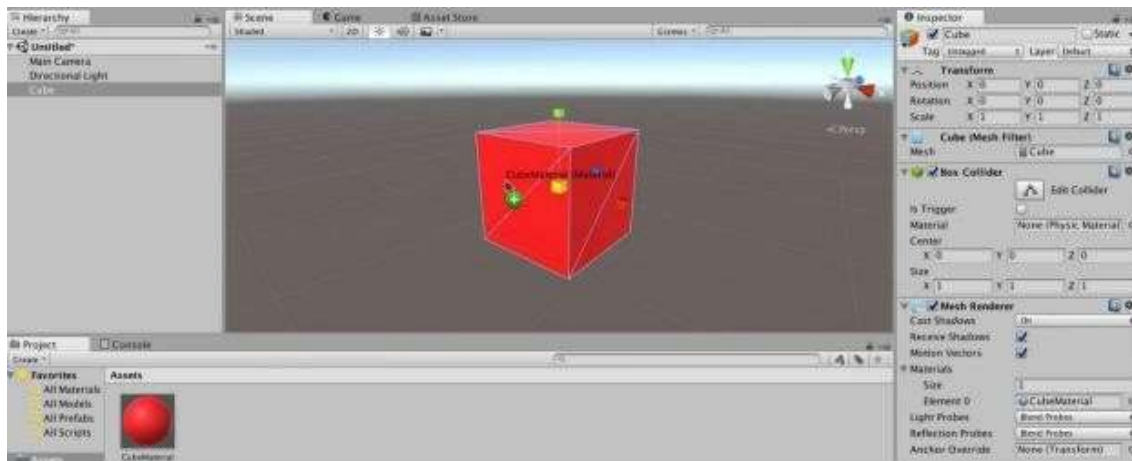
Što je više crvene boje, veća je vrijednost crvene boje. Isto vrijedi za zelenu i plavu.

Alfa se odnosi na prozirnost boje. Čisto crvena boja ima vrijednosti crvene i alfa vrijednost 255. Vrijednosti plave i zelene su 0. Da bismo ColorMaterialu dali čistu crvenu boju, prikazano je na slici ispod:



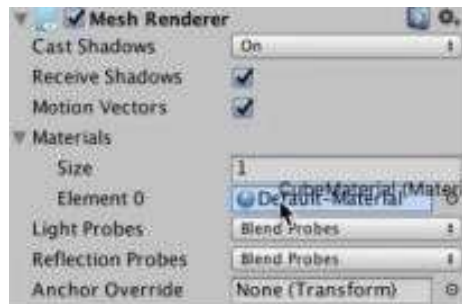
Slika 28 ColorMaterial

Da biste Kocki dodijelili ovu crvenu boju, povucite i ispustite CubeMaterial na kocku u Sceni.



Slika 29 CubeMaterial kocke

Alternativno, povucite i ispustite CubeMaterial u polje Element 0 u Cube's Mesh Renderer komponenta. Boja kocke postat će crvena.,

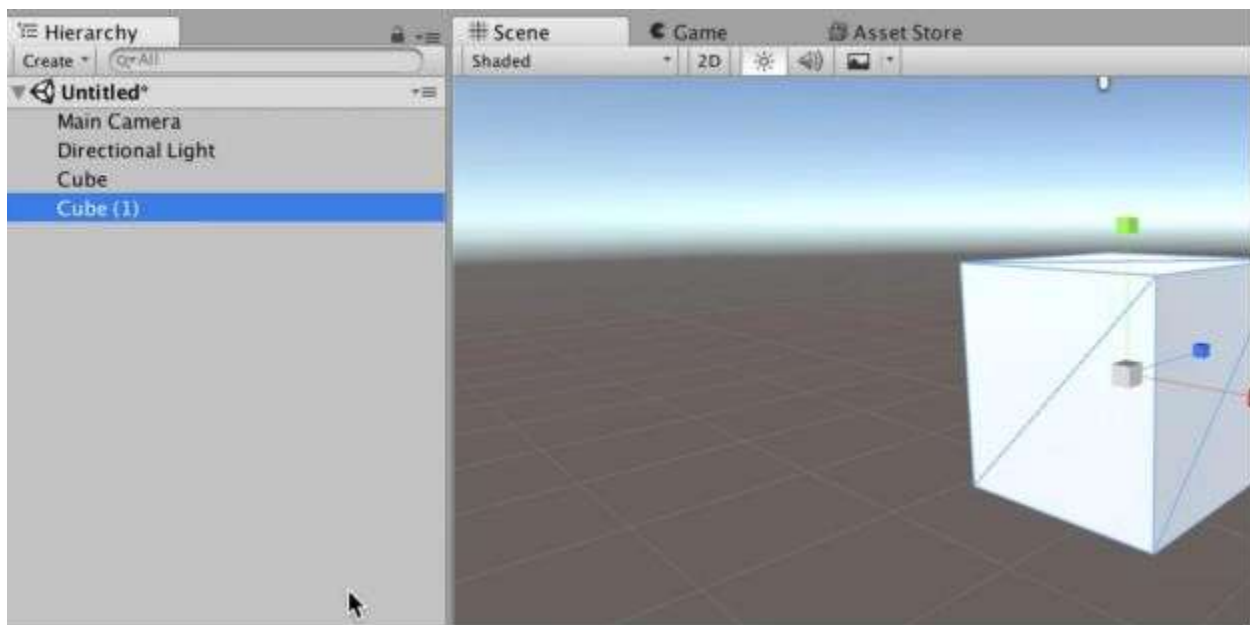


Slika 30 CubeMaterial

Primjena transparentnosti

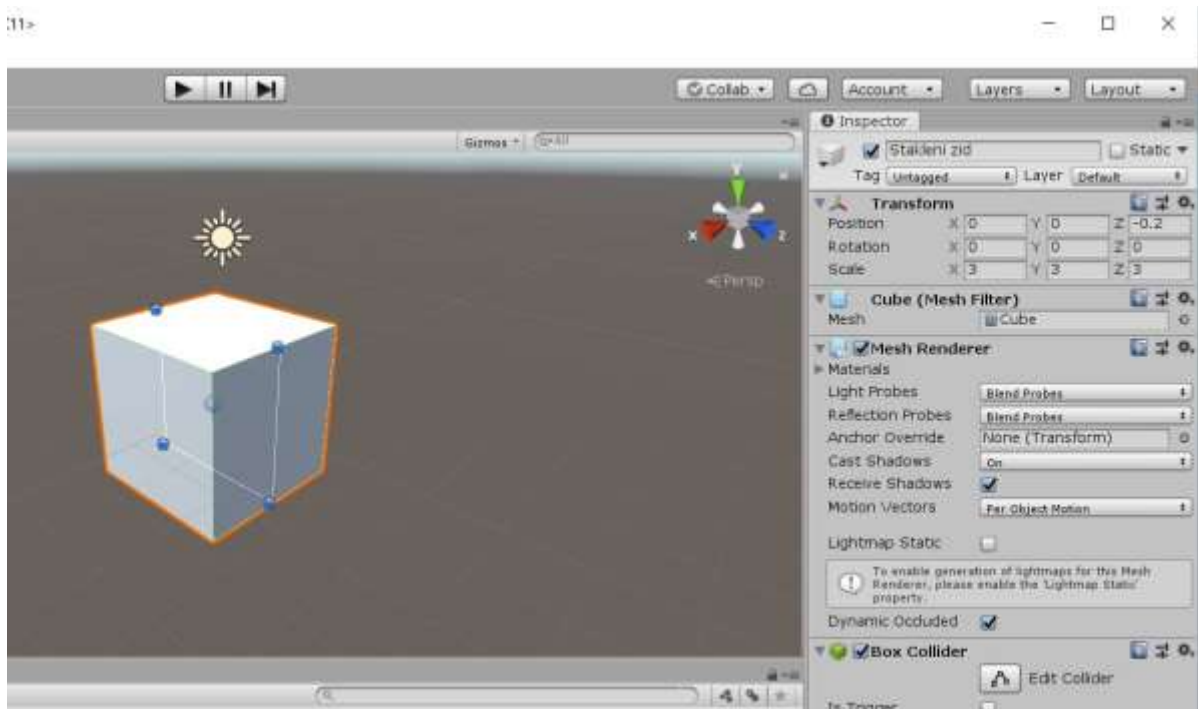
Što ako želimo da kocka bude prozirna? Odaberite CubeMaterial. Odaberite polje za boje pokraj "Albeda" u Inspektoru. Pokušajte promijeniti vrijednost alfa. Kockin izgled neće se promijeniti. Promjena RGB vrijednosti objekta ažurira boju kocke. Isto ne vrijedi za promjenu prozirnosti predmeta.

Da bismo promijenili alfa objekta, napravimo zid koji ćemo učiniti poluprozirnim. Desnom tipkom miša kliknite Hijerarhiju. Odaberite 3D Object> Cube da biste stvorili novu kocku, koja će se zvati "Cube (1)".



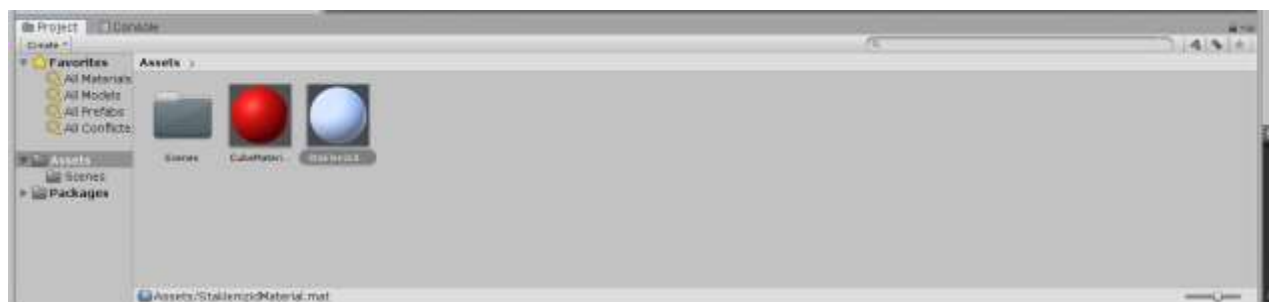
Slika 31 Transparentnost kocke

U Inspektoru promijenite X-položaj kocke na 0, Y-položaj na 0 i Z-položaj na -1. Skalirajte Kocku na vrijednosti kocke X,Y, Z = 3. Preimenujte kocku u "Stakleni zid".



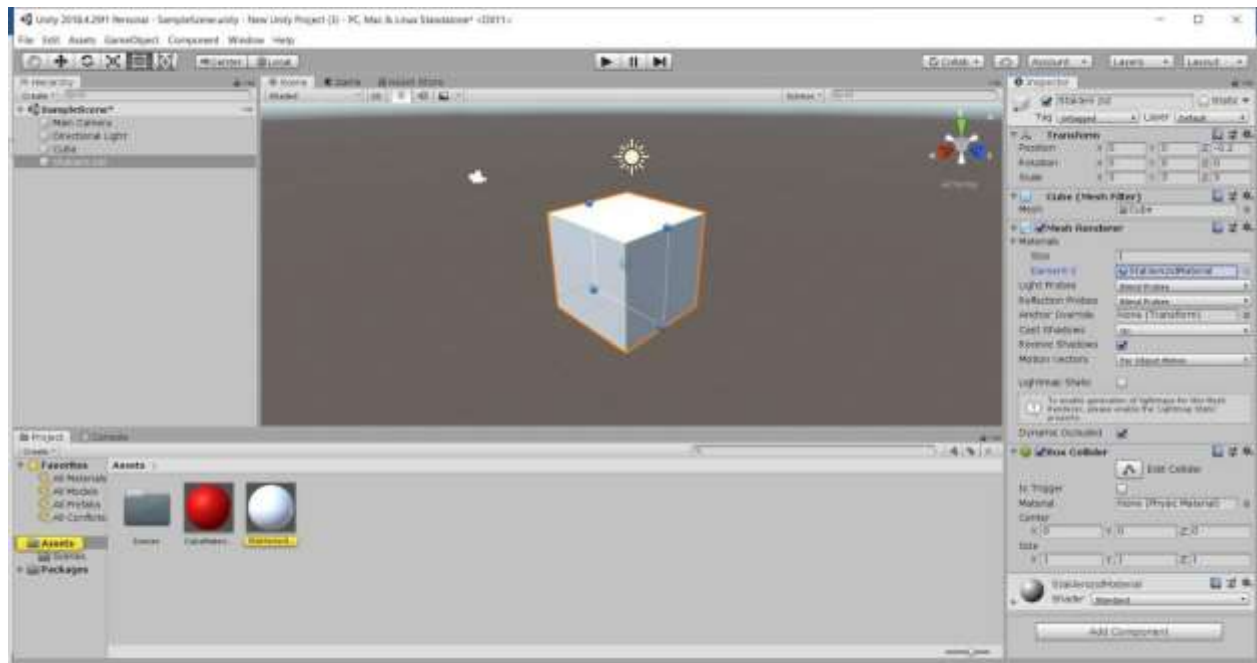
Slika 32 Preimenovanje kocke

Napravimo materijal za "Stakleni zid". Desnom tipkom miša pritisnite u prozoru Projekt. Odaberite Create> Materijal za stvaranje novog materijala. Nazovite materijal ""StaklenizidMaterijal".



Slika 33 Materijal Stakleni Zid

Da biste primijenili StaklenizidMaterial na Staklenizid, odaberite materijal i povucite ga do Elementa 0 polje u komponenti Mesh Renderer.



Slika 34 Postavljenje materijala na kocku

Odaberite StaklenizidMaterial. Promijenite boju materijala u plavu. Smanjite alfa vrijednost na polovicu zadane vrijednosti.



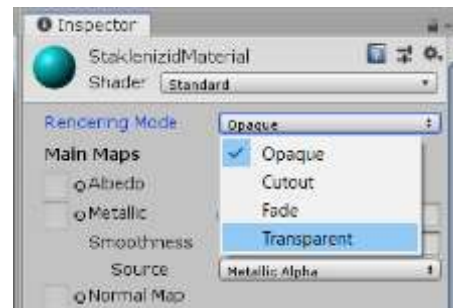
Slika 35 Podešavanje transparentnosti

Transparentnost je teško postići na taj način. Stoga trebate posebno reći renderu da se želi postići transparentnosti.

U inspektoru za materijal StaklenizidMaterial -a promijenite:

Način prikazivanja iz "Neprozirnog/Opaque" u "Prozirni/Transparent".

Transparentnost StaklenizidMaterial -a promijenit će se u Sceni.

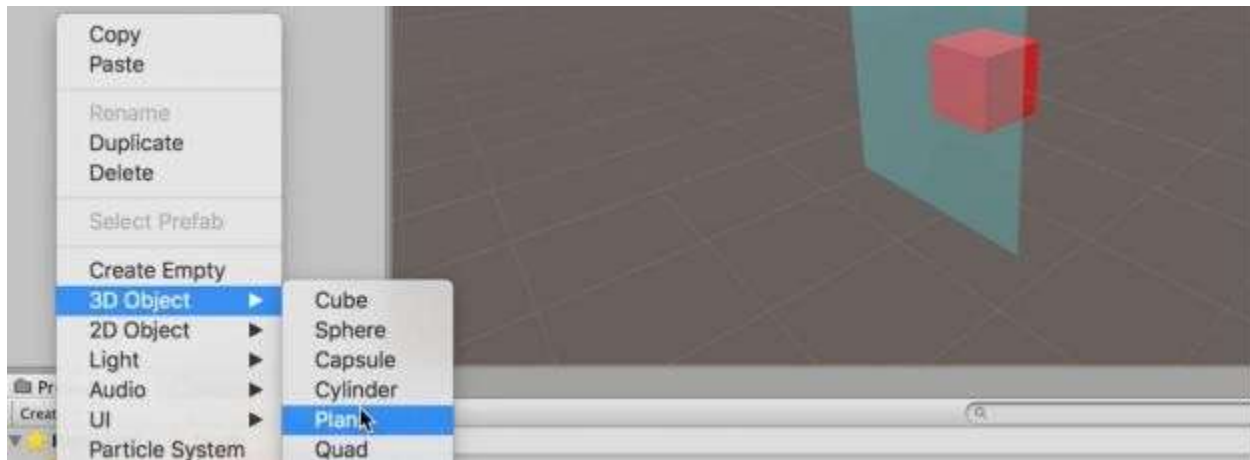


Slika 36 Postavljanje Redering Mode

Da bi promjena transparentnosti bila vidljivija, vi može smanjiti alfa vrijednost StaklenizidMaterial -a.

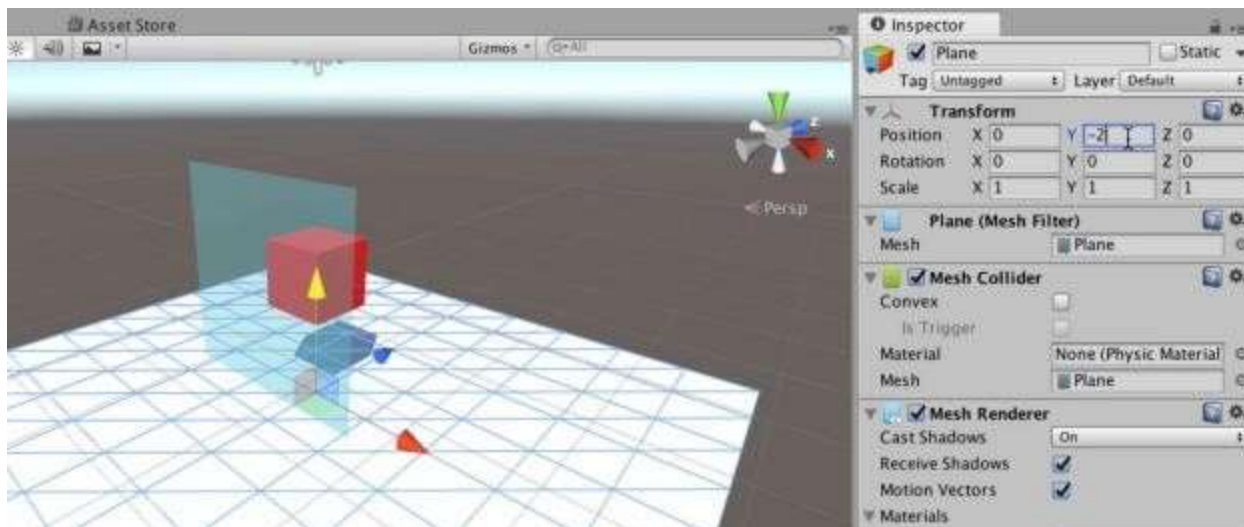
4.2. Izrada Plohe

Izrada podloge odnosno tla za igru. Desnom tipkom miša kliknite Hijerarhiju i odaberite 3D Object> Plane.



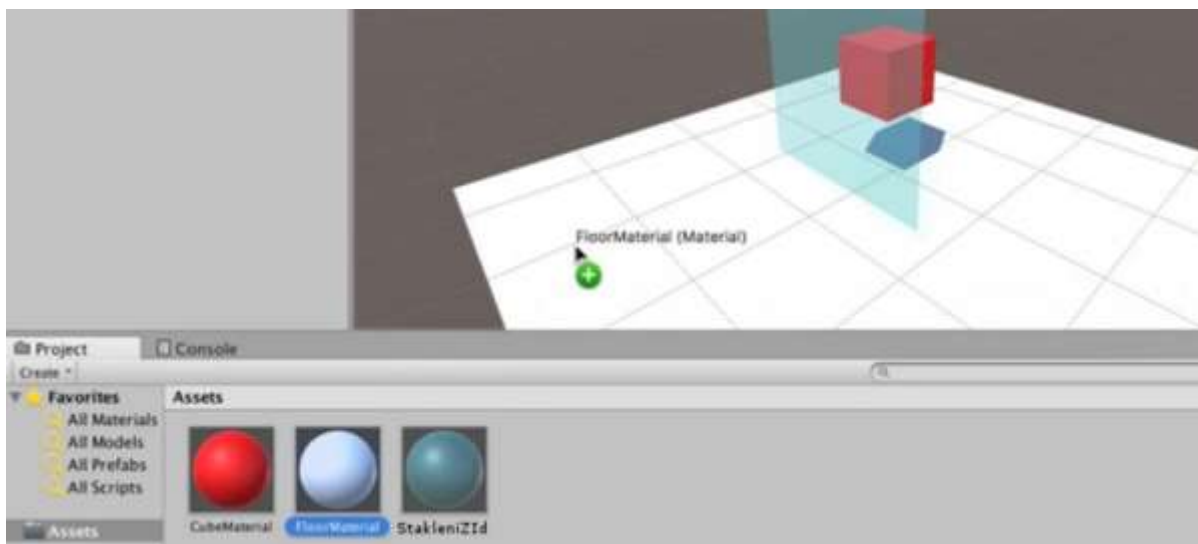
Slika 37 Stvaranje 3D objekta

Kao što je vidljivo na sljedećoj slici, u Prizoru će se pojaviti tanka ravnina. Stavimo položaj 0 -2 0.



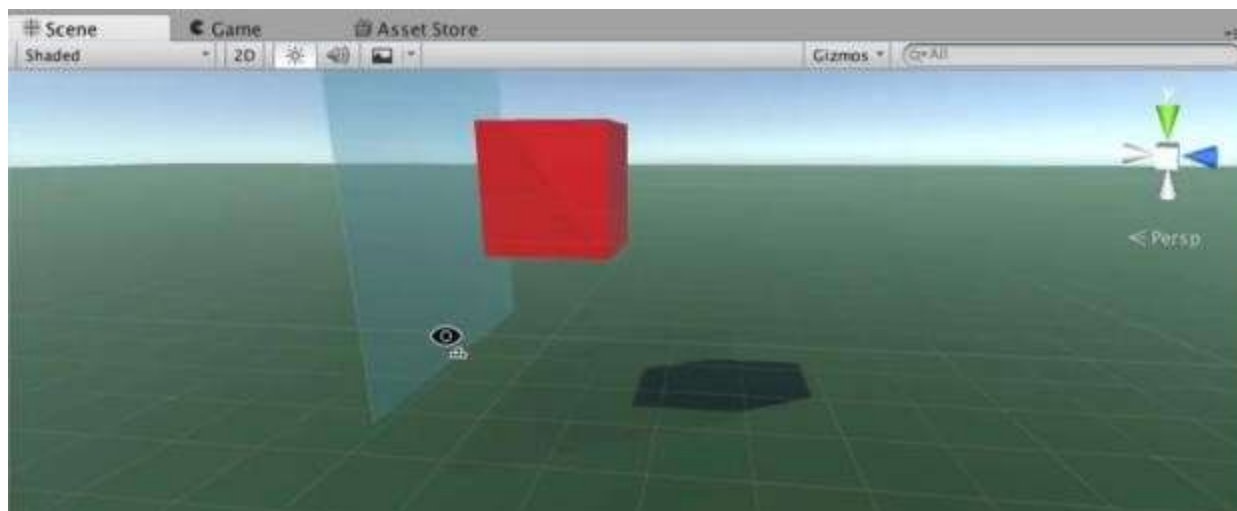
Slika 38 Izrada podloge

Da biste promijenili boju plohe, stvorite novi materijal u programu Assets. Nazovite materijal "FloorMaterial ". Nanesite materijal na Plohu na Sceni.



Slika 39 Stavljanje materijala na podlogu

U Inspectoru promijenite boju Plane (podloge) u tamno zelenu s RGB vrijednostima kao što je 15 39 12. Da biste povećali veličinu ravnine, napravite vrijednosti X i Z scale na 20. Moći ćete vidjeti sjene prikazane u Sceni. Sjena kocke projicirat će se na plohu.



Slika 40 Prikaz sjene u Sceni

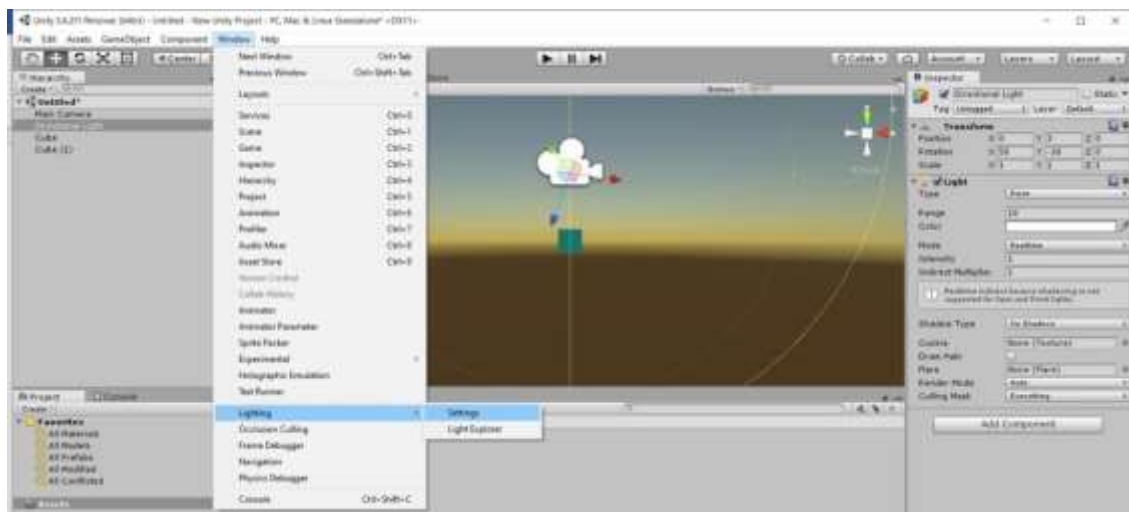
Promijenite Y položaj Y na -1,5. Kocka će potonuti kroz pod.

3. Svijetlost

Osvjetljenje je bitan dio razvoja igre. Zadano svijetlo Unity-a je Usmjereno svjetlo. Zrake se šire od usmjerenog svjetla u svim smjerovima. Kad zraka pogodi objekt, sjena se stvara na podu.

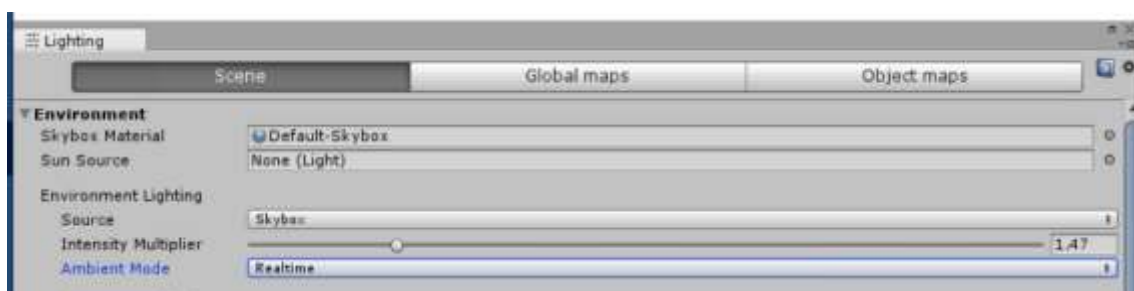
Intenzitet ambijenta

Jedan od načina za promjenu osvjetljenja igre je promjena intenziteta ambijenta Scene. Odaberite Usmjereno svjetlo (eng. Directional Light). Resetirajte komponentu Light u Inspektoru. Iz izbornika Window odaberite Light->Settings



Slika 41 Svijetlost u igrama

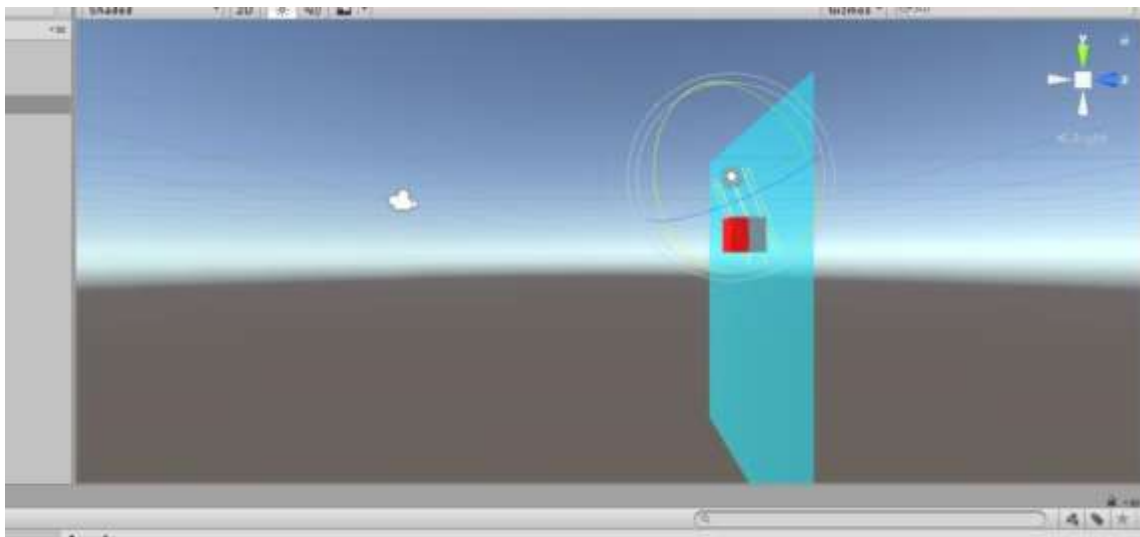
Pojavit će se prozor Lighting. Odaberite karticu Scene. Ovdje možete promijeniti vrijednost klizač "Intensity Multiplier". Izgled Scene promijenit će se ovisno o vrijednosti.



Slika 42 Postavke ambijenta u igrama

Postavite Intenzitet ambijenta na 1. Zatvorite prozor Osvjetljenje. Promijenite položaj Directional Light po bilo kojoj osi. Primijetite da intenzitet svjetla Prizora i sjene se ne mijenjaju. To je zato što je Directional Light toliko daleko da ne utječe na projekciju.

Međutim, **promjena rotacije Directional Light mijenja rasvjetu scene**. Odaberite alat Rotation u gornjem lijevom kutu. U Sceni pomičemo krugove Directional Light. Izgled sjena u Sceni će se promijeniti.

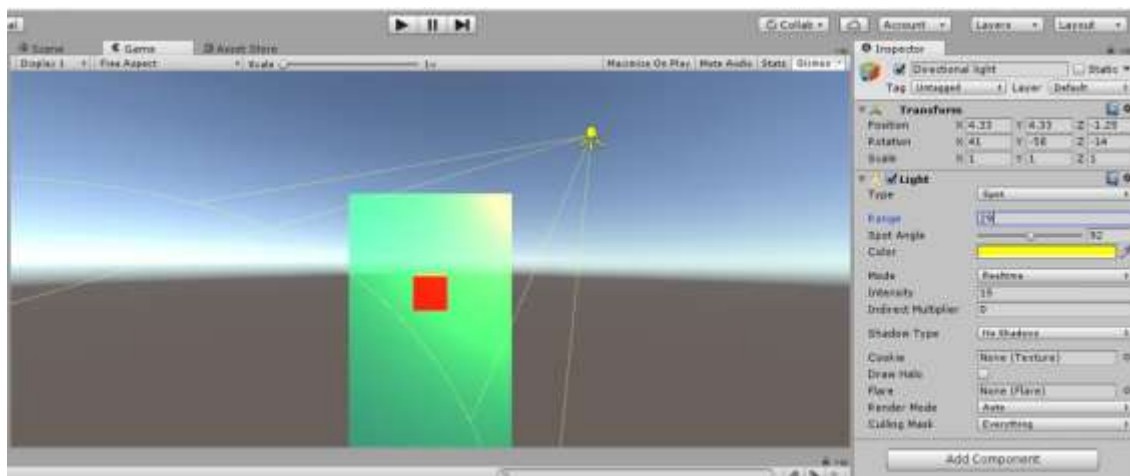


Slika 43 Rotiranje svjetlosti u Sceni

Rotiranje usmjerene svjetlosti je poput pomicanja sunca. Upućujući svjetlo u potpunosti iznad poda Prizor čini mračnim, kao da je noć.

Spot Light

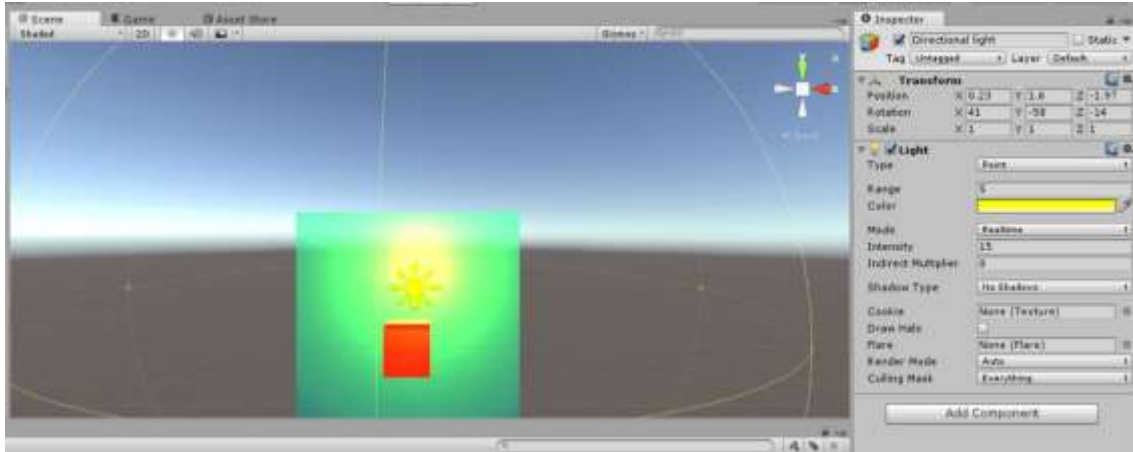
Postoje i druge vrste svjetla. U komponenti Svjetlost u Inspektoru promijenite Type svjetlosti na Spot. **Spot je reflektor koji se usmjerava na određeno područje**. Može se promijeniti domet reflektora (koliko daleko pokazuje), njegov kut (koliko je širok) i njegov intenzitet.



Slika 44 Spot svjetlost

Točkasta svjetlost

Druga vrsta svjetlosti je Točka (*eng. Point*), prikazana na sljedećoj slici. **Point svjetlo je svjetlo žarulja.** To je točka koja projicira svjetlost u svim smjerovima.



Slika 45 Point svjetlost

Žuta kupolu predstavlja domet točkaste **Point** svjetlosti.

Postoje mnoga druga svojstva koja možete izmijeniti kako biste promijenili osvjetljenje scene, pa ostanite znatiželjni i eksperimentirajte u Inspektoru.

4. Roll-A.Ball - uvod u tutorijal

Game engine koji će biti korišten u ovom tutorijalu je Unity 3D. Prikazati će se korak po korak stvaranje mini igre Roll-A-Ball. Verzija programa Unity, koja će se koristiti, je 5.5.1f1.

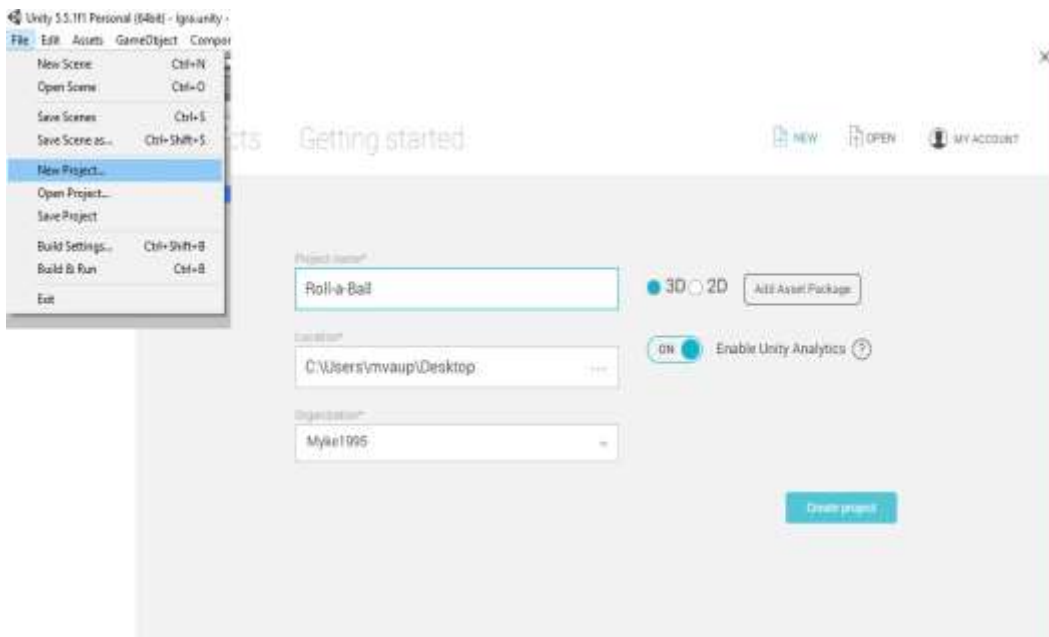
Roll-A-Ball je jednostavna igra namijenjena za početnike koja pokazuje princip rada sa objektima igre, komponentama, fizikom te skriptama. Prikazati će se kako stvoriti novi objekt, kako dodati nove komponente objektima igre, dodijeliti im svojstva i vrijednosti te kako ih pozicionirati u scenu igre. Koristit će se jednostavni geometrijski oblici, odnosno tijela bez uporabe dodatnih sredstava, modela, tekstura, zvukova ili animacija. Na kraju će se dobiti jednostavna igra u kojoj igrač upravlja kuglom, pomoću tipkovnice. Kugla se kotrlja po ravnoj ploči koristeći fiziku i silu. Korištenjem ulaznih vrijednosti određenih tipki sa tipkovnice zadavati će se sila kugli te je tako tjerati da se kreće po sceni. Prikazati će se kako uspostaviti kontakt sa kuglom i objektima koje je moguće sakupiti, te iskoristiti te događaje kako bi se sakupljali objekti igre te tako povećao rezultat. Igra završava kada igrač sakupi sve objekte na ploči.

5. Igrač i okolina

Da bi se igra postavila, najprije treba izraditi novi projekt. Projekt će se zvati Roll-a-Ball. Putanja za spremanje projekta uzima se po izboru, za početak projekt se može spremiti na radnu površinu. S obzirom da je igra 3D, kod stvaranja projekta mora se odabrati opcija 3D.

Koraci:

1. Otvoriti izbornik File
2. Odabrati New project
3. Otvora se novi prozor, pod project name upisati Roll-a-Ball
4. Upisuje se putanja spremanja projekta pod Location, za početak projekt spremiti na radnu površinu
5. Odabrati opciju 3D
6. Klik miša na Create project

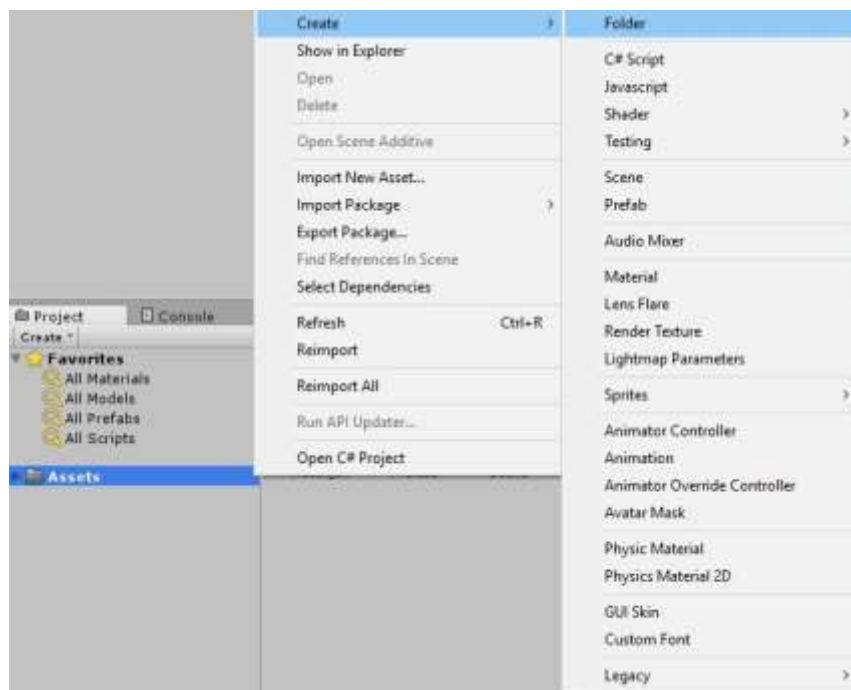


Slika 46 Izrada novog projekta

Prva stvar koju treba napraviti kada se napravi novi projekt je spremiti trenutnu scenu. Bitno je da su sve datoteke i materijali uredno organizirani radi lakšeg snalaženja kasnije. Zato treba izraditi novu mapu u mapi Assets i nazvati je „Scene“. Scenu nazvati „Igra“ te ju spremiti u mapu Scene.

Koraci:

1. U prozoru Project desnim klikom na mapu Assets odabire se opcija Create ->Folder



Slika 47 Stvaranje nove mape u mapi Assets

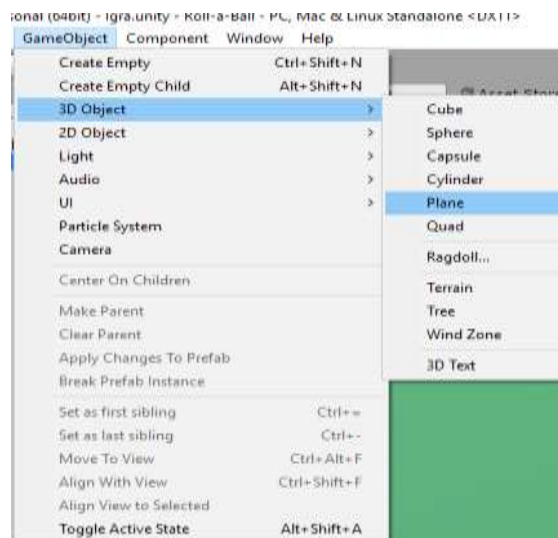
2. Duplim lijevim klikom na ime novo-dodane mape – promjeniti ime na „Scene“
3. Otvoriti izbornik File
4. Odabrati Save Scenes as..
5. Scenu nazvati Igra te je spremiti u podmapu Scene u mapi Assets

Kugla se neće kretati po praznom prostoru, stoga treba dodati ploču po kojoj će se kretati. Preko izbornika GameObject odabire se novi standardni Unity objekt zvan Plane. Plane je obična prazna ploha koja označava površinu. Površina će se zvati „Ploča“. Za slučaj da ploča nije centrirana u sceni, mora se resetirati na način da joj se koordinate pozicije x,y i z postave na vrijednost 0.

Pomoću alata za mijenjanje veličine ploče moguće je lako podesiti njezine dimenzije. Taj alat se pojavljuje na sceni kada se klikne na neki objekt scene, u ovom slučaju je to „Ploča“. Isto se može učiniti malo preciznije ako se upišu vrijednosti za koordinate u Transform Scale. Unity kao zadane vrijednosti veličine koristi 1x1 „Unity jedinica“. S obzirom da ploča nema visinu, ne može se podesiti po y osi. Kako bi se jasnije vidjeli objekti na površini, ploči je moguće postaviti drugu boju pozadine. Boja pozadine je materijal, stoga treba napraviti novu mapu u koju će se spremati svi ostali materijali. U mapu Assets treba dodati mapu Materijali te u nju dodati novi materijal. Taj materijal zvat će se „Pozadina“. Nakon imenovanja materijala preostaje samo postaviti željenu boju. Gotovi materijal jednostavno se povuče na ploču te ona poprima boju materijala.

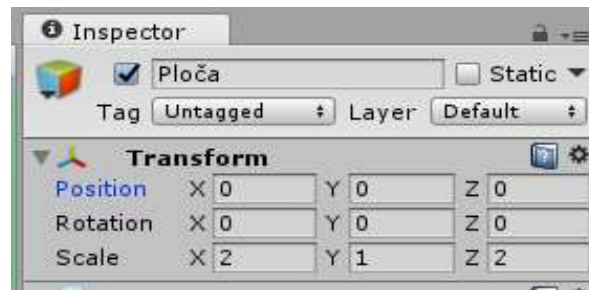
Koraci:

1. Otvoriti izbornik GameObject na alatnoj traci
2. Odabrati izbornik 3D Object->Plane



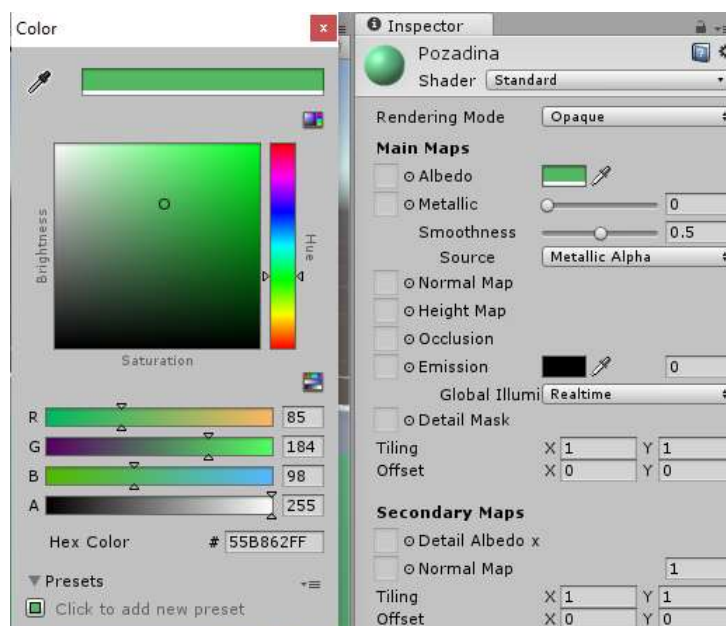
Slika 48 Stvaranje novog 3D objekta "Plane"

3. Preimenovati objekt u „Ploča“
4. Odabrati objekt „Ploča“
5. Na desnoj strani nalazi se prozor „Inspector“
6. U prozoru Inspector, pod Transform, postaviti Position koordinate za na 0



Slika 49 Transform komponenta objekta "Ploča"

7. Dodati novu mapu, nazvati ju „Materijali“, Create->Folder
8. Desni klik na mapu materijali, Create->Material
9. Preimenovati materijal u „Pozadina“
10. Odabrati materijal
11. U prozoru „Inspector“ odabrati opciju „Albedo“
12. Odabrati željenu boju pozadine
13. Povučemo materijal na ploču na sceni



Slika 50 Postavljanje boje materijala

Nakon izrade početne površine na nju treba dodati i kuglu preko izbornika GameObject. Dodanu kuglu nazvati "Igrač". Ako koordinate kugle već nisu postavljene na 0, to je prvo što treba učiniti. Centar kugle je sada točno u sredini scene. Želimo da kugla stoji na plohi. S obzirom da je zadana veličina kugle 1x1, treba postaviti y koordinatu pozicije na 0.5, pa će kugla savršeno doticati plohu..

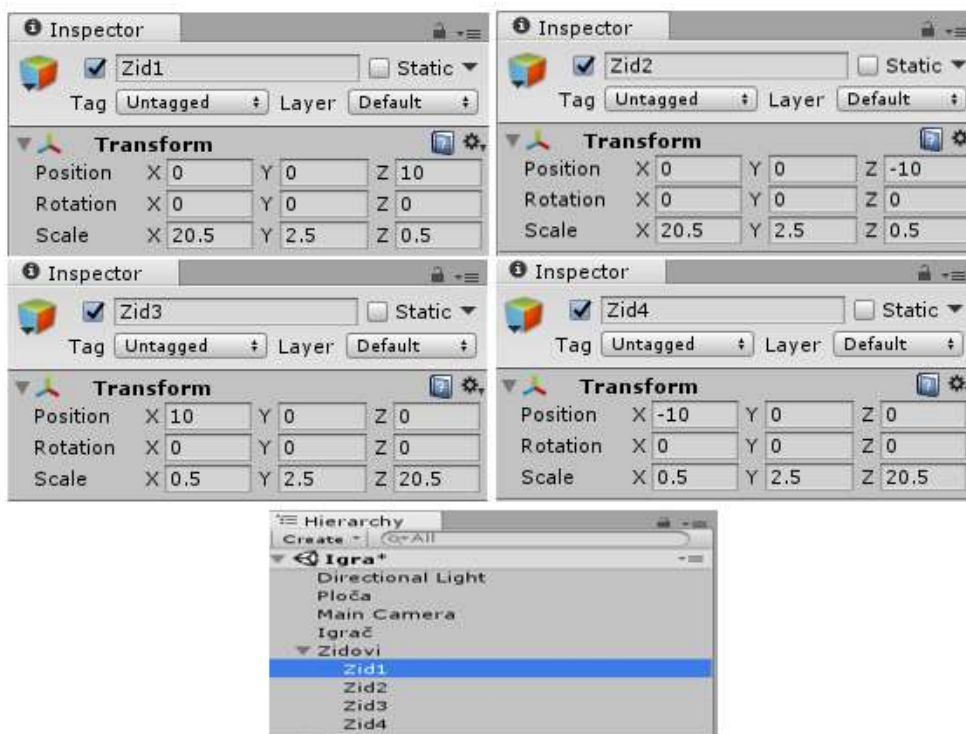
Koraci:

1. Otvoriti izbornik GameObject
2. Odabrati 3D-Object->Sphere
3. Preimenovati Sphere u „Igrač“
4. U prozoru Inspector postaviti Position koordinate na 0, y koordinatu postaviti na 0,5

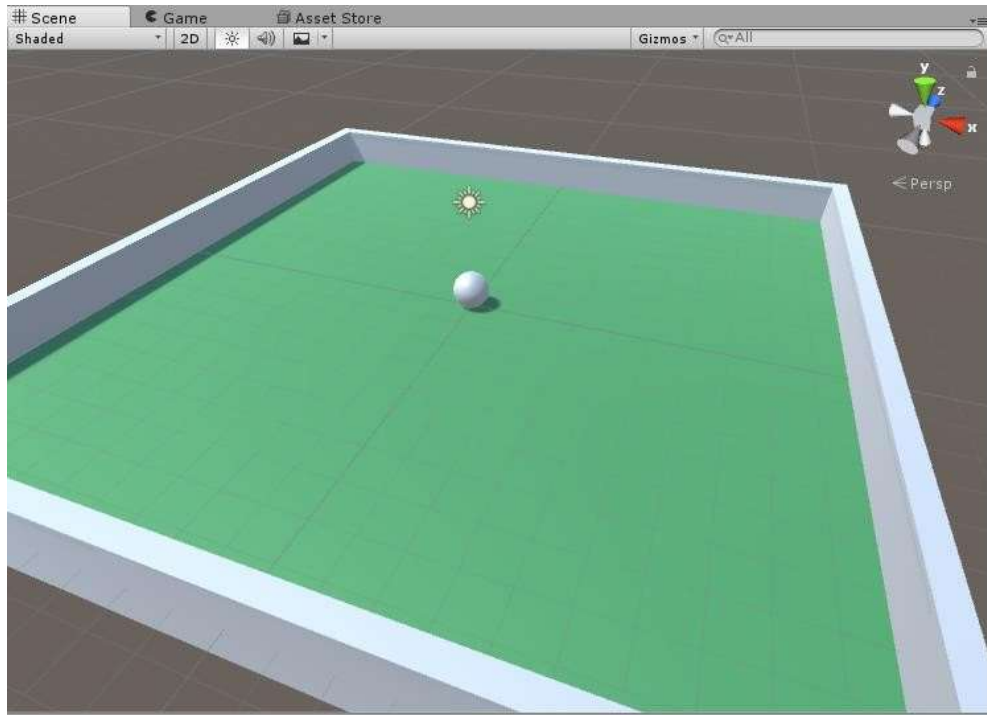
Kada su igrač i ploča postavljeni preostaje još dodati i zidove koji će sprječavati kuglu da padne sa ruba. Za to će biti potrebno 5 individualnih objekata koji će se ujediniti u jedan objekt igre. Prvo treba dodati jedan prazan objekt koji će se zvati „Zidovi“. Zatim dodati objekt Cube koji će biti jedan dio zida. Upisuju se koordinate i dimenzije zida tako da se prilagodi duljini ploče te pozicionira na jednu stranu ploče. Potrebna su četiri takva zida, po jedan za svaku stranu ploče, pa ga stoga treba kopirati još 3 puta. Dimenzije ostalih zidova ostaju iste, no mijenja im se rotacija i pozicija tako da je na kraju po jedan zid sa svake strane. Te zidove treba ujediniti povlačenjem na prazan objekt Zidovi u pogledu scene. Time su objekti zidova postali djeca objekta Zidovi. Svrha ujedinenja objekata u jedan prazan objekt u ovom slučaju služi za uredno organiziranje objekata u pogledu scene što kasnije omogućuje lakše snalaženje. Nakon ovih koraka okolina za igru je postavljena.

Koraci:

1. Otvoriti izbornik GameObject
2. Odabrati Create empty
3. Prazan objekt nazvati „Zidovi“
4. Ponovno otvoriti izbornik GameObject
5. Odabrati 3D-Object->Cube
6. Objekt preimenovati u „Zid1“
7. Objekt Zid1 kopirati i zalijepiti 3 puta
Odabire se objekt, Ctrl+C, nakon toga 3 puta Ctrl+V
8. Objekte redom nazvati „Zid2“, „Zid3“ te „Zid4“
9. „Odabrati objekt, te u prozoru Inspector unijeti koordinate i vrijednosti za svaki pojedini zid:
10. Povuci svaki pojedini objekt zida na objekt „Zidovi“



Slika 51 Koordinate za postavljanje objekata zidova



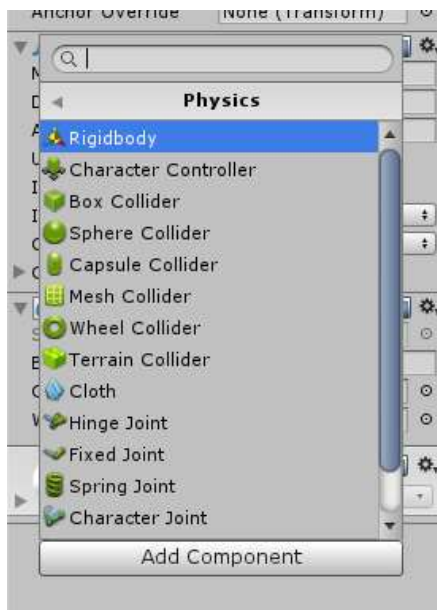
Slika 52 Izgled scene nakon postavljanja

5.1. Pomicanje igrača

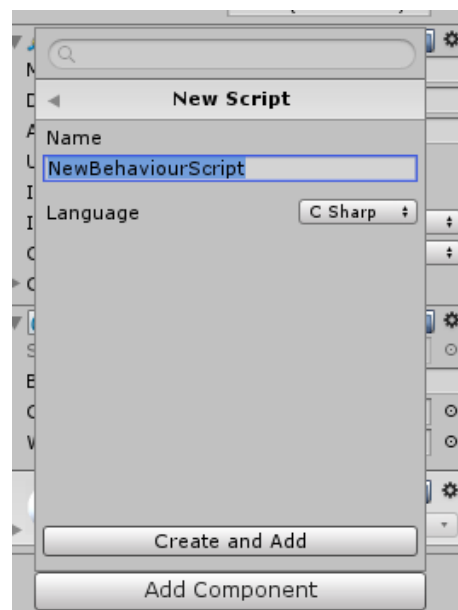
Nakon postavljanja početne okoline igrača, potrebno je implementirati kretanje igrača po ploči. Za to je potrebno dodavanje sile na kuglu. Da bi se dodala sila na kuglu potrebno je koristiti unos sa tipkovnice. Potrebne su dvije komponente koje treba dodati objektu; Rigidbody komponenta, te skripta koja će upravljati komponentom Rigidbody i unosom sa tipkovnice. Rigidbody komponenta omogućuje dodavanje fizike na objekt. Radi organizacije, stvoriti novu mapu u koju će se spremati sve skripte. Programski jezik skripte biti će C#.

Koraci:

1. Odabrati objekt Igrač
2. U Inspector prozoru kliknuti na gumb Add Component na dnu prozora
3. Odabrati Physics->RigidBody
4. Desni klik na mapu Assets, Create->Folder
5. Novu mapu preimenovati u „Skripte“
6. Ponovno odabrati objekt Igrač
7. Add Component->New Script
8. Skriptu nazvati „Kretanje“
9. Odabrati jezik skripte „C Sharp“
10. Kliknuti na gumb Create and Add
11. Skriptu povući iz Assets mape u mapu Skripte



Slika 53 Dodavanje Rigidbody komponente



Slika 54 Dodavanje nove skripte

U dodanu skriptu potrebno je napisati programski kod. Prvo treba otvoriti skriptu u Visual Studio-u. Ono što skripta treba raditi je provjeravati svaki okvir (eng. *frame*) za unos sa tipkovnice te taj unos pretvoriti u silu na kuglu. Kod za fiziku piše se u blok naredbi `FixedUpdate`. Taj blok se izvodi prije svake kalkulacije fizike. Kugla će se kretati samo horizontalno i vertikalno odnosno ne će se kretati u visinu, pa stoga joj treba mijenjati samo koordinate x i z. U blok `FixedUpdate` dodaju se dvije nove varijable koje će dobiti vrijednost preko funkcije `Input.GetAxis`. Funkcija Uzima vrijednosti horizontalne i vertikalne osi koje su kontrolirane tipkama na tipkovnici. Tipke za unos moguće je podesiti po želji, no Unity zadane tipke za horizontalne i vertikalne osi su w,a,s i d ili tipke strelica. Kako bi se dodala sila na `Rigidbody` komponentu potrebno je koristiti `Vector3` klasu. `Vector3` označava da sila koristi 3 vektora, odnosno daje silu u 3 smjera; smjer x, y te z. Kugla se miče horizontalno i vertikalno tako da joj se na vrijednost x i z koordinati pozicije dodaju vrijednosti prethodno dodanih varijabli, dok je y vektor postavljen na 0. Potrebne su dvije dodatne varijable; privatna varijabla tipa `Rigidbody` za pristup `Rigidbody` komponenti kugle te javna varijabla sa kojom će se mijenjati brzina kretanja. Preostaje još dodati silu na `Rigidbody`, te u blok naredbi `Start` dodati naredbu za dohvaćanje `Rigidbody` komponente što omogućuje da se skripta pokreće sa pokretanjem igre.

Programski kod skripte:

```
using UnityEngine;
using System.Collections;

public class Kretanje : MonoBehaviour {

    //Deklaracija varijable za brzinu te Rigidbody komponentu
    public float speed;
    private Rigidbody rb;

    void Start ()
    {
        //dohvaćanje komponente
        rb = GetComponent<Rigidbody>();
    }

    void FixedUpdate ()
    {
        //dohvaćanje vrijednosti za osi
        float kretanjeHorizontal = Input.GetAxis("Horizontal");
        float kretanjeVertical = Input.GetAxis ("Vertical");

        //postavljanje vektora
        Vector3 kretanje = new Vector3 (kretanjeHorizontal,
        0.0f, kretanjeVertical);

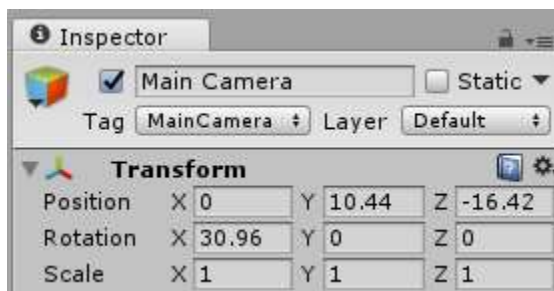
        //Dodavanje sile na objekt
        rb.AddForce (movement * speed);
    }
}
```


5.2. Postavljanje kamere

Najbitniji objekt scene je kamera. Bez kamere se igra može pokrenuti, ali na ekranu ne će biti prikazano ništa. Zbog toga Unity automatski dodaje objekt glavne kamere kod stvaranja projekta. Tu kameru treba i postaviti. Ona se treba kretati zajedno sa igračem radi lakšeg pogleda i snalaženja u igri. Kako bi se kamera kretala zajedno sa objektom igrača, na nju je potrebno dodati skriptu koja će je povezati sa igračem te je pomicati u skladu s njime.

Koraci:

1. Odabrati objekt Main Camera u hijerarhijskom pregledu objekata scene
2. Postaviti vrijednosti pozicije i rotirati kameru tako da ima dobar pogled na igrača. Za početak koristiti ove vrijednosti:



Slika 55 Transform vrijednosti za postavljanje kamere

3. Kliknuti na gumb Add Component
4. Dodati novu C# skriptu zvanu KameraController

Programski kod skripte treba sadržavati javnu varijablu koja će služiti kao referenca na objekt Igrač. Osim toga, treba dodati privatnu varijablu tipa Vector 3 koja će mijenjati poziciju kamere u odnosu na vrijednost Vector3 varijable iz skripte igrača. Mora se uzeti u obzir da je pozicija kamere različita od pozicije igrača, stoga pozicija kamere je trenutna pozicija igrača plus razlika pozicije igrača te kamere. Nakon napisanog koda, objekt Igrač se povezuje sa kamerom tako da se jednostavno povuče u komponentu skripte na kameri.

Programski kod KameraController-a:

```
public class KameraController : MonoBehaviour {

    public GameObject player;

    private Vector3 pomak;

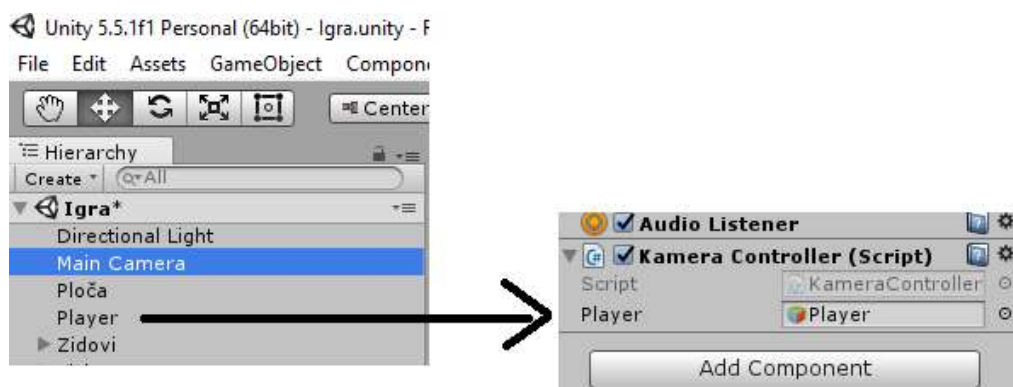
    void Start ()
    {
        pomak = transform.position - player.transform.position;
    }

    void LateUpdate ()
    {
        transform.position = player.transform.position + pomak;
    }

}
```

Povezivanje objekta igrača sa skriptom kamere:

-objekt povući na dio Player komponente skripte



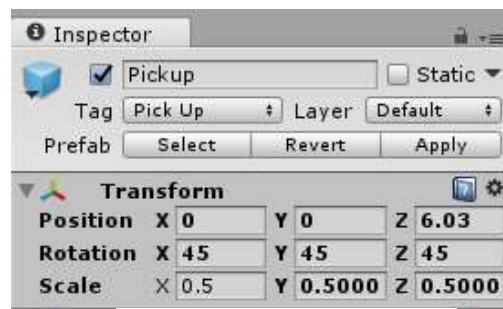
Slika 56 Povezivanje kamere sa skriptom objekta Igrač

6. Sakupljanje objekata i prikaz rezultata

Igrač pokreće kuglu po ploči i ne može pasti s nje, no da bi igra imala smisla treba imati neki cilj. Dodavanjem objekata koji se mogu sakupiti igri se daje svrha. Prvo treba dodati jedan objekt te ga oblikovati, a kasnije svi ostali objekti se mogu samo kopirati. Objekt neka se zove Pick Up. Objekti za sakupljanje se moraju nekako isticati na površini kako bi lakše upali u oko igraču. Potrebno je stvoriti novi objekt kocku, smanjiti je i rotirati tako da ostavlja dojam kao da lebdi u zraku. Animacija vrtnje te promjena boje postići će da se što više ističe na ploči. Rotacija objekta radi se preko skripte. Svaki naknadni objekt biti će isti kao i prvi. Kako se kasnije ne bi svaki objekt morao mijenjati svaki zasebno, objekt treba pretvoriti u „prefab“. Prefab je element koji sadrži postavke objekta. Stvaranjem prefaba omogućuje se da svi objekti istog prefaba imaju ista svojstva, pa ako se promjeni jedan objekt, promjenit će se svi ostali. Radi organizacije, prefab treba spremići u novu mapu. Nakon što je prvi objekt gotov, treba kopirati i rasporediti ostale objekte te ih na kraju grupirati u jedan novi prazan objekt.

Koraci:

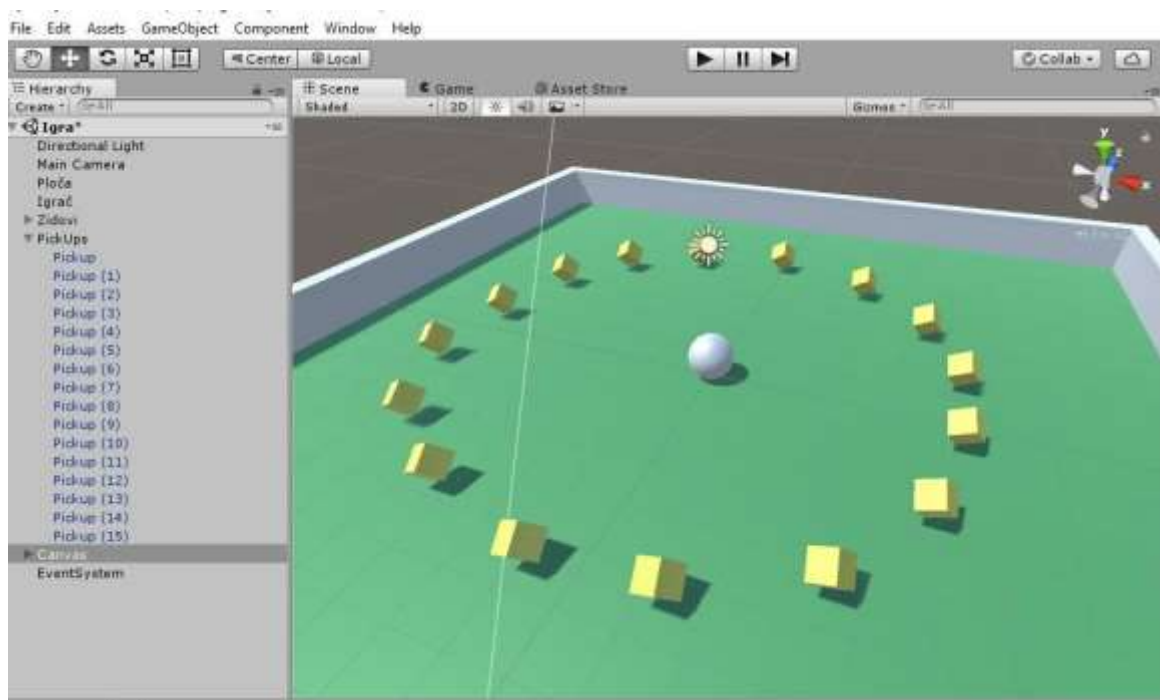
1. Otvoriti izbornik GameObject
2. Dodati novi 3D-Object Cube
3. Imenovati ga Pick Up
4. Odabrati taj objekt
5. U prozoru Inspect unijeti vrijednosti za rotaciju, te smanjiti veličinu svih dimenzija na 0.5



Slika 57 Transform koordinate objekta za sakupljanje

6. Dodati novu C# skriptu, Add Component->New Script, nazvati je Rotacija
7. Skriptu povući iz Assets mape u mapu Skripte
8. Desnim klikom na mapu Assets dodati novu mapu, Create->Folder
9. Mapu nazvati Prefabs

10. Pickup objekt povuči u mapu Prefabs
11. Dodaje se novi prazan Game Object, nazvati ga „Pick Ups“
12. Pick Up objekt povuči na objekt Pick Ups
13. Objekt kopirati te ga zalijepiti 15 puta
14. Objekte rasporediti oko igrača u krug, ili po želji
15. Desni klik na mapu Materijali, dodati novi materijal, Create->Material
16. Materijal nazvati „PickUp“
17. Odabrati materijal Pickup
18. Desno u prozoru inspector odabrati Albedo opciju
19. Odabrati željenu boju, različitu od boje ploče
20. Materijal Pickup povući na prefab Pick Up u mapi Prefabs



Slika 58 Izgled scene sa postavljenim objektima

Programski kod skripte Rotacija:

```
public class Rotator : MonoBehaviour {  
  
    void Update ()  
    {  
        //svaki frame se rotacija objekta mijenja  
        transform.Rotate (new Vector3 (15, 30, 45)*Time.deltaTime);  
    }  
}
```

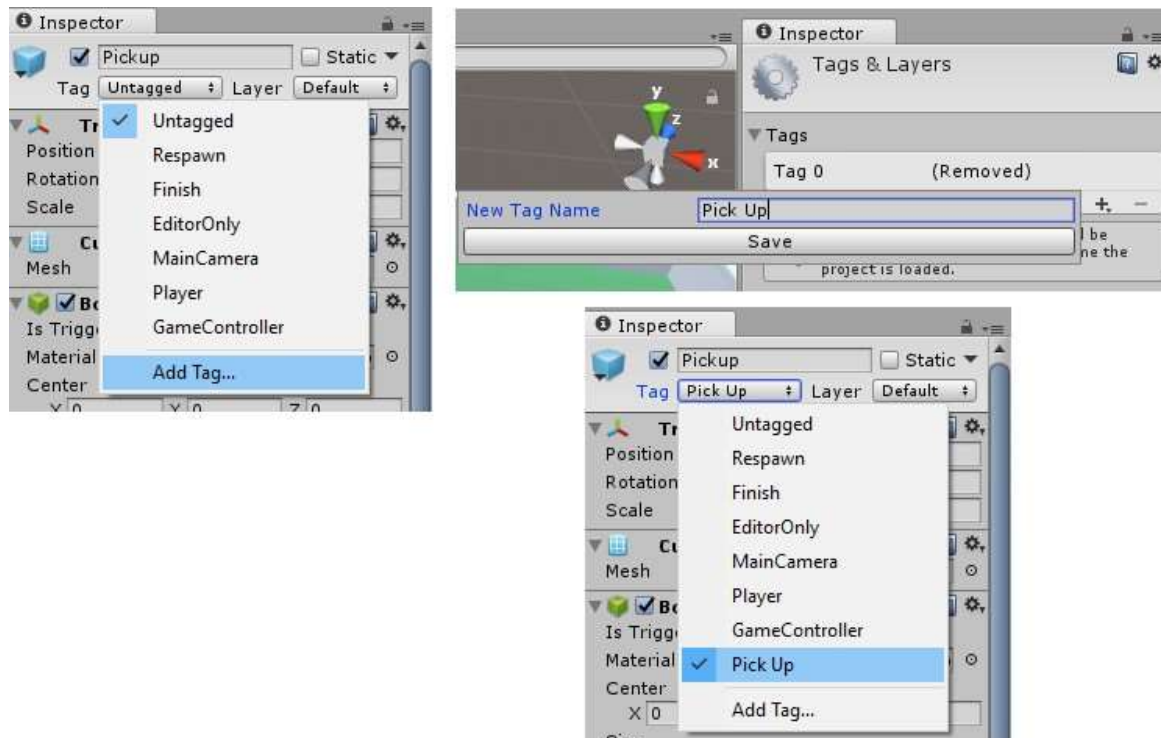
6.1. Sakupljanje objekata

Postavljene objekte igrač treba nekako sakupiti. Kako bi ih sakupio, potrebno je otkriti dodir igrača i objekta za sakupljanje. Kada dođe do kontakta objekta i igrača potrebno je sakriti ili uništiti objekt. To se ostvaruje pomoću skripte. Skriptu Kretanje, koja je povezana sa objektom igrača, je potrebno doraditi. Svaki objekt igre ima prepoznavanje sudara (eng. *Collider*). Kako igrač ne bi slučajno sakupio zid ili samu ploču, treba testirati te identificirati koji collider je dotaknut. Samo kada je Igrač u kontaktu sa collider-om objekta za sakupljanje, tada taj objekt može sakupiti. U skriptu Kretanje dodaje se nova metoda *OnTriggerEnter* tipa void koja se poziva kada igrač uspostavi kontakt sa nekim drugim objektom. Metoda kao parametar uzima collider objekta s kojim dolazi u kontakt. Ako collider drugog objekta ima oznaku „Pick Up“, taj objekt postaje neaktivan. Oznaka (eng. *Tag*) „Pick Up“ ne postoji ni na jednom objektu trenutno, stoga ju je potrebno dodati na objekte koje želimo sakupiti. Potrebno je odabrati prefab objekta Pick Up te u prozoru Inspector dodati novi tag koji će se zvati „Pick Up“. Kako se igrač ne bi odbijao od objekata za sakupljanje, potrebno je collider Pick Up objekta pretvoriti u trigger collider. Kada je omogućen trigger collider, ostali objekti mogu prelaziti kroz Pick Up objekte, stoga objekti mogu pasti sa ploče jer ih sila gravitacije povuče. To se izbjegne na način da se objekti definiraju kao Kinematic objekt. Kada je objekt kinematičan, sile ne djeluju na njega.

Koraci:

1. Odabire se prefab Pick Up
2. U prozoru Inspector klikuti na padajući izbornik Tag
3. Odabrati Add Tag...

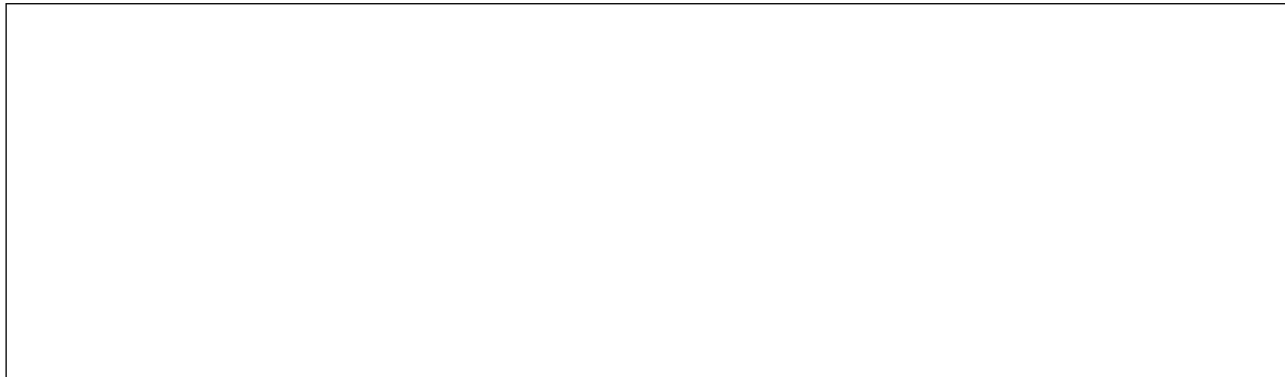
4. Kliknuti na gumb + u dijelu Tags
5. Nazvati novi tag „Pick Up“
6. Kliknuti na gumb Save
7. Ponovno odabrati prefab Pick Up
8. U prozoru Inspector odabrati padajući izbornik Tag
9. Odabrati tag Pick Up



Slika 59 Dodavanje oznake na objekt Pick Up

10. Odabrati Pick Up prefab
11. U prozoru Inspector, u dijelu Box Collider, označiti kvačicom Is Trigger
12. U dijelu Rigidbody označiti kvačicom Is Kinematic

Programski kod koji je potrebno dodati u skriptu Kretanje:

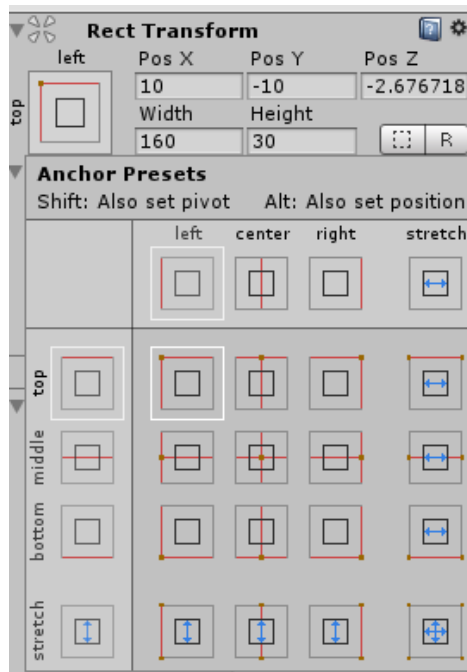


6.2. Prikaz rezultata na korisničkom sučelju

Zadnja stvar koju treba izraditi u igri je korisničko sučelje koje će prikazivati rezultat. Potrebno je dodati novi UI objekt Text. Odabirom objekta Text stvaraju se dva dodatna objekta; Canvas te Event System. Svaki element korisničkog sučelja je dijete objekta Canvas. Zbog toga se Canvas automatski dodaje. Najprije objekt Canvas treba povezati sa kamerom. Zatim postaviti tekst u gornji lijevi kut Canvasa. Nakon toga treba ponovno nadograditi skriptu Kretanje programskim kodom koji će povezati text objekt sa skriptom te omogućiti promjenu teksta sakupljanjem objekata. Kod se sastoji od jedne javne varijable koja služi kao referenca na Text objekt, od jedne privatne varijable u koju se sprema rezultat, te metode SetCountText koja ispisuje tekst u Text objekt. Nakon što je to obavljeno, mini-igra je gotova.

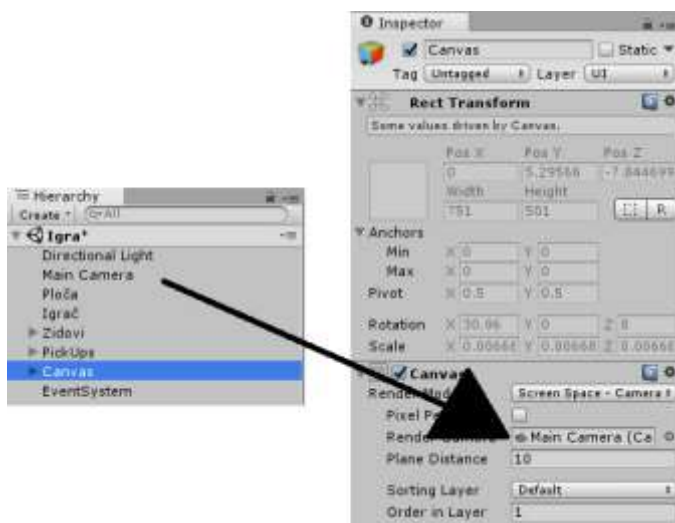
Koraci:

1. Otvoriti izbornik GameObject
2. Odabrati UI->Text
3. Odabrati objekt Text
4. U prozoru Inspector u dijelu Rect Transform Postaviti Pos X na 10, Pos Y na -10
5. Kliknuti na kvadratić u gornjem lijevom kutu Rect Transform-a
6. Uz držanje tipki Shift+Alt odabrati poziciju teksta u gornji lijevi kut



Slika 60 Postavljanje pozicije Text objekta na Canvasu

7. Odabrati objekt Canvas
8. Povuci objekt Main Camera na Render Camera u Canvas dijelu prozora Inspector



Slika 61 Povezivanje kamere sa korisničkim sučeljem

9. Upisati programski kod u skriptu

Cijeli kod skripte Kretanje:

```
using UnityEngine;
using UnityEngine.UI;
using System.Collections;

public class Kretanje : MonoBehaviour {

    public Text countText;
    public float speed;
    private int zbroj;
    private Rigidbody rb;

    void Start ()
    {
        //dohvaćanje komponente
        rb = GetComponent<Rigidbody>();
        count = 0;
        SetCountText ();
    }

    void FixedUpdate ()
    {
        float kretanjeHorizontal = Input.GetAxis("Horizontal");
        float kretanjeVertical = Input.GetAxis ("Vertical");
        Vector3 kretanje = new Vector3 (kretanjeHorizontal,
            0.0f, kretanjeVertical);

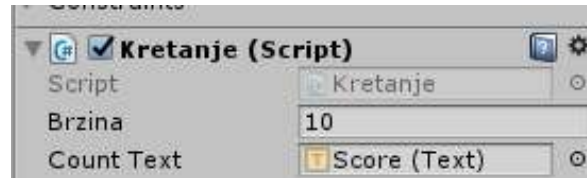
        rb.AddForce (movement * speed);
    }

    void OnTriggerEnter(Collider other)
    {
        if (other.gameObject.CompareTag ("Pick Up"))
        {
            other.gameObject.SetActive (false);

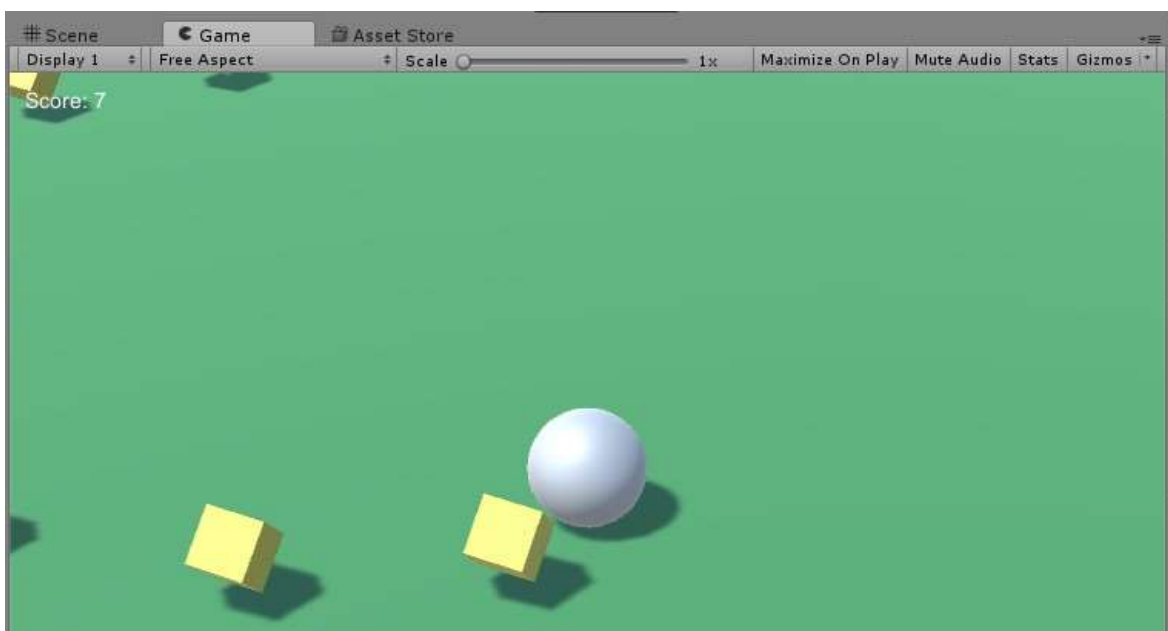
            //nakon svakog sakupljenog objekta
            // rezultat se povećava za 1
            zbroj += 1;
            SetCountText ();
        }
    }
    //postavlja tekst objekta Text
    void SetCountText ()
    {
        countText.text = "Rezultat: " + zbroj.ToString ();
    }
}
```

10. Odabire se objekt Igrač

11. U prozoru Inspector, povuči objekt Text na Count Text varijablu komponente skripte Kretanje



Slika 62 Ulazne varijable za skriptu Kretanje



Slika 63 Izgled gotove pokrenute igre

Ovim tutorialom prikazano je jednostavno stvaranje mini-igre korištenjem Unity 3D Game Engine-a. Prikazani su osnovni koraci dodavanja novih objekata na scenu, oblikovanja objekata pomoću materijala i Transform alata te međusobno povezivanje objekata. Također obrađena je osnova skriptiranja u Unity-u i korištenje Physics Engine-a. Tutorijal je namijenjen za apsolutne početnike i vrlo je dobar uvod u izradu računalnih igara.

7. Popis literature

1. Roll-a-Ball Tutorial. <https://learn.unity.com/tutorial/environment-and-player?projectId=5c51479fedbc2a001fd5bb9f> (15.5. 2021.)
2. Unity Manual verzija 5.5.
<https://docs.unity3d.com/550/Documentation/Manual/UnityManual.html> (19.5.2021.)
3. FELICIA, Patrick Unity from Zero to proficiency (Beginner) : A step-by-step guide to coding your first game / Patrick Felicia. - 3rd ed. - Wroclaw : Patrick Felicia, 2019.
4. THORN, Alan Unity 5.x By Example : An example-based practical guide to get you up and running with Unity 5.x / Alan Thorn. - 1st ed. - Birmingham : Packt Publishing, 2016.

8. Popis slika

Slika 1 Početna stranica Unity.com	4
Slika 2 Unity store	4
Slika 3 Početni zaslon	5
Slika 4 Izgled Unity Editora	5
Slika 5 Izgled Scene	6
Slika 6 Izgled Main Camere	7
Slika 7 Camera Preview	7
Slika 8 Izgled Game scene	8
Slika 9 Izgled Direction Light	8
Slika 10 Inspector prozor	9
Slika 11 Inspector glavne kamere	9
Slika 12 Transform glavne kamere	9
Slika 13 Postavke kamere	10
Slika 14 Postavke svijetla	10
Slika 15 Assets prozor	10
Slika 16 Prozor Console	11
Slika 17 Postavljanje pozicije kamere	12
Slika 18 Postavljanje pozicije kamere	12
Slika 19 Rotiranje kamere	14
Slika 20 Stvaranje kocke	15
Slika 21 Kocka u Sceni	15
Slika 22 Skaliranje kocke	16
Slika 23 skaliranje kocke	16
Slika 24 Box Collider kocke	17
Slika 25 Izrada Materijala	18
Slika 26 Materijal kocke	18
Slika 27 Main Maps	19
Slika 28 ColorMaterial	20
Slika 29 ColorMaterial kocke	20
Slika 30 ColorMaterial	21
Slika 31 Transparentnost kocke	21
Slika 32 Preimenovanje kocke	22
Slika 33 Materijal Stakleni Zid	22
Slika 34 Postavljanje materijala na kocku	23
Slika 35 Podešavanje transparentnosti	23
Slika 36 Postavljanje Redering Mode	24
Slika 37 Stvaranje 3D objekta	24
Slika 38 Izrada podloge	25
Slika 39 Stavljanje materijala na podlogu	25
Slika 40 Prikaz sjene u Sceni	26
Slika 41 Svjetlost u igrama	27
Slika 42 Postavke ambijenta u igrama	27

Slika 43 Rotiranje svjetlosti u Sceni.....	28
Slika 44 Spot svjetlost	28
Slika 45 Point svjetlost.....	29
Slika 46 Izrada novog projekta	31
Slika 47 Stvaranje nove mape u mapi Assets	32
Slika 48 Stvaranje novog 3D objekta "Plane"	33
Slika 49 Transform komponenta objekta "Ploča"	34
Slika 50 Postavljanje boje materijala	34
Slika 51 Koordinate za postavljanje objekata zidova	36
Slika 52 Izgled scene nakon postavljanja	37
Slika 53 Dodavanje Rigidbody komponente	38
Slika 54 Dodavanje nove skripte	38
Slika 55 Transform vrijednosti za postavljanje kamere	41
Slika 56 Povezivanje kamere sa skriptom objekta Igrač.....	42
Slika 57 Transform koordinate objekta za sakupljanje	43
Slika 58 Izgled scene sa postavljenim objektima	44
Slika 59 Dodavanje oznake na objekt Pick Up.....	46
Slika 60 Postavljanje pozicije Text objekta na Canvasu.....	48
Slika 61 Povezivanje kamere sa korisničkim sučeljem	48
Slika 62 Ulazne varijable za skriptu Kretanje	50
Slika 63 Izgled gotove pokrenute igre	50